

Krylov-Unterraum basierte Modellreduktion zur Simulation von Werkzeugmaschinen

Von der Carl-Friedrich-Gauß-Fakultät
der Technischen Universität Carolo-Wilhelmina zu Braunschweig

zur Erlangung des Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigte

D i s s e r t a t i o n

von Dipl.-Math. Andreas Soppa,

geboren am 12.12.1975,

in Wolfsburg

Eingereicht am: 17.06.2011

Mündliche Prüfung am: 06.12.2011

Referentin: Professor Dr. Heike Faßbender, TU Braunschweig

Korreferent: Professor Dr.-Ing. Michael Zäh, TU München



Soppa, Andreas:

*Krylov-Unterraum basierte Modellreduktion zur
Simulation von Werkzeugmaschinen,*

Dissertation,

Technische Universität Carolo Wilhelmina,

Braunschweig, 2012.

Kurzfassung

Die vorliegende Arbeit hat die Reduktion von Strukturmodellen, wie sie bei der Simulation von Werkzeugmaschinen zum Einsatz kommt, zum Thema. Dabei steht die Entwicklung neuer mathematischer Reduktionsverfahren, die auf Krylov-Unterräumen basieren und sich durch eine Automatisierbarkeit auszeichnen im Fokus der Arbeit. Daneben soll jedoch auch die Bedeutung der Modellreduktion als wichtiges Hilfsmittel zur Gewährleistung einer effizienten Simulation im Gesamtentwicklungsprozess von Werkzeugmaschinen dargestellt werden.

Moderne Werkzeugmaschinen sind heute komplexe mechatronische Produktionssysteme, deren Entwicklung unter großem Innovations-, Zeit- und Kostendruck stattfindet. Aus diesem Grund erlangen Techniken, die den Entwicklungsprozess unterstützen, zunehmend an Bedeutung. Dabei ist der Einsatz solcher Techniken umso effektiver, je früher im Produktentwicklungsprozess wichtige Erkenntnisse über das Verhalten eines Gesamtsystems gewonnen werden können. Dabei besteht das Ziel die Realisierbarkeit neuer Funktionen und die Erreichbarkeit der Qualitäts- und Leistungsanforderungen bereits in der Entwurfsphase sicherstellen zu können. Die Simulationstechnik stellt in diesem Zusammenhang eine Schlüsseltechnologie zur frühzeitigen Überprüfung des Verhaltens von Produktionsanlagen zur Verfügung.

Ein besonders effizienter und frühzeitig anwendbarer Ansatz zur Abschätzung der erzielbaren Leistungsfähigkeit und Genauigkeit von Maschinenentwürfen ist die Kopplung von Finite-Elemente-Modellen der mechanischen Struktur mit der regelungstechnischen Simulation der elektrischen Antriebssysteme. Dabei werden erste Entwurfsmodelle der mechanischen und elektrischen Konstruktion sowie der Steuerungstechnik mit geringfügigem Zusatzaufwand für die Modelldatenverarbeitung in ein mechatronisches Gesamtmodell integriert.

Soll das Verhalten einer Maschine mit einer guten Auflösung simuliert werden, führt dies zu sehr hochdimensionalen Modellen. Für eine effiziente Berechnung des Gesamtsystems, insbesondere für die Zeitbereichssimulation, ist es dann notwendig, dass die Anzahl der Freiheitsgrade, die durch die Diskretisierung der Finite-Elemente-Methode bestimmt wird, deutlich reduziert wird. Hierfür wurden bislang die sogenannten modalen Verfahren verwendet, die auf der Lösung des Eigenwertproblems beruhen, um damit das Originalsystem mit Hilfe der Modalmatrix auf einen deutlich kleineren, modalen Raum zu projizieren.

Da mit diesen Verfahren eine optimale Approximation lediglich durch die Berechnung sämtlicher Eigenwerte erreicht werden kann, was zu sehr langen Berechnungszeiten führt, werden lediglich vom Anwender gewählte oder die niederfrequenten Eigenwerte für die Ordnungsreduktion verwendet. Dies hat zur Folge, dass die entstehenden reduzierten Modelle zum Teil erhebliche Fehler

im statischen Bereich aufweisen. Die modalen Verfahren berücksichtigen lediglich das System alleinstehend, es erfolgt demnach keine Beachtung der Systemein- bzw. -ausgänge, die vor allem für die gekoppelte Simulation mit Modellen der regelungstechnischen Komponenten einer Werkzeugmaschine von Bedeutung sind. Zudem existieren keine Schranken für den Approximationsfehler, wodurch die Modellreduktion mit diesen Verfahren schwer zu automatisieren ist.

Neue Ansätze zur Modellreduktion mit Hilfe mathematischer Verfahren, welche gegenüber den konventionellen Methoden zahlreiche Vorteile aufweisen, lassen ein erhebliches Potenzial erkennen das Originalsystem besser approximieren zu können. Zudem lassen sich diese automatisieren und weisen über ein breiteres Frequenzband sehr gute Approximationseigenschaften auf. Diese Verfahren weisen auch die nötige Effizienz auf, um zukünftige Maschinengenerationen bereits am virtuellen Prototypen zuverlässig testen zu können. Neben dem erkennbaren Potential dieser Verfahren treten bei ihrer praktischen Anwendung jedoch Fragen bezüglich der optimalen Auslegung der zu wählenden Parameter der Algorithmen auf.

Im Fokus der vorliegenden Arbeit liegt die Betrachtung neuer Ansätze zur Modellreduktion, die auf Krylov-Unterräumen basieren. Dabei besteht das Ziel dieser Arbeit in der Entwicklung neuer Reduktionsverfahren, die auf der Grundlage mathematischer Verfahren zur Modellreduktion aufbauen und eine Automatisierung des Modellreduktionsprozesses ermöglichen. Dazu werden neue Reduktionsverfahren vorgestellt und auf ihre Prozesstauglichkeit hinsichtlich der modellreduzierten Simulation von Werkzeugmaschinen validiert, die eine auf die besonderen Strukturmodelle von Werkzeugmaschinen angepasste und automatisierte Modellreduktion der Finite-Elemente-Modelle ermöglichen, die aus einer Modellierung rotatorischer Antriebssysteme mit Kugelgewindetrieb im Verbund mit dem Maschinengestell resultieren.

Die neu entwickelten Verfahren basieren dabei auf einer angepassten iterativen und adaptiven Auswahl der für die Verfahren benötigten Parameter. Zudem basieren diese Verfahren auf globale Krylov-Unterräume und der Verwendung der globalen Arnoldi-Methode, die sich durch eine hohe Effizienz bei der Berechnung der Reduktion von Systemen mit mehreren Ein- und Ausgängen auszeichnet. Mit Hilfe einer geeigneten Methode zur Abschätzung des Approximationsfehlers des reduzierten Systems erlauben die neu entwickelten Verfahren eine automatische Modellreduktion ohne Benutzerinteraktion während des Reduktionsprozesses.

Die Ergebnisse der vorliegenden Arbeit basieren auf Ergebnissen, die im Rahmen eines von der Deutschen Forschungsgemeinschaft geförderten Projektes mit der Bezeichnung *WAZorFEM: Integrierte Simulation des Systems "Werkzeugmaschine-Antrieb-Zerspanprozess" auf Grundlage ordnungsreduzierter FEM-Strukturmodelle* erarbeitet wurden. Dieses Projekt hatte unter anderen Arbeitspunkten die Untersuchung der Anwendbarkeit von Krylov-Unterraum-Verfahren zur Reduktion von Modellen mechanischer Strukturen, wie sie bei der Simulation von Werkzeugmaschinen auftreten, als Bestandteil.

Abstract

The subject of the present work is the reduction of structural models, as used in the simulation of machine tools. The focus of the work is the development of new mathematical reduction methods, based on Krylov-subspaces and distinguished by a feasible automation. In addition, however, the importance of model reduction as an important tool for an efficient simulation in the overall development process of machine tools is presented.

Modern machine tools are complex mechatronic production systems. Their development takes place under innovation, time, place and cost pressures. Hence, techniques to support the development process, become increasingly important.

The use of such techniques is more effective the earlier in the development process the behaviour of the production systems can be obtained. Ideally, the feasibility of new features and the accessibility of quality and performance standards are guaranteed at the design phase. The simulation technique in this context is a key technology for early verification of the behavior of production systems.

A particularly effective and applicable approach to early obtain achievable productivity and accuracy of NC machines is the coupling of finite element models of the mechanical structure with the control engineering simulation of electric drive systems. For this, first draft models of the mechanical structure, the electrical construction and the control were integrated into a mechatronic model with slight additional effort for the model data processing.

If the behavior of a machine is simulated with a good resolution, this results in very high-dimensional models. For an efficient calculation of the integrated system, particularly for time domain simulation, it is necessary that the number of degrees of freedom, which is determined by the discretization of the finite element method, is significantly reduced. For this purpose usually the so-called modal method based on the solution of the eigenvalue problem is used to project the original system using the modalmatrix to a significantly smaller modal space.

With the modal method an optimal approximation can only be achieved by calculation of all eigenvalues which leads to very long calculation times. Hence, only user specified or the low eigenvalues are used for the order reduction. As a result, the resulting reduced models have considerable errors in the static area.

Moreover the modal method considers the system only with no notice of the system in- and outputs and there are no bounds for the approximation error.

New mathematical approaches to model order reduction which offer many advantages, compared

to modal reduction indicate a significant potential approximating the original system better. In addition, methods based on this approaches can be automated and have better approximation properties on a wider band of frequencies. These methods also have the necessary efficiency, to test the virtual prototypes of future generations of machines.

In this work, adapted mathematical procedures for automated model reduction of finite element models, obtained by the modeling of the rotary drive systems with ball screw in association with the machine frame, are developed and validated.

The procedures are based on global Krylov-subspaces and the global Arnoldi-method characterized by high efficiency in the calculation of the reduction of systems with multiple inputs and outputs. Moreover, these methods allow a model reduction without user interaction.

The present work is based on results in the framework of a project supported by the German Research Community (DFG). Subject of the project was to investigate the applicability of Krylov subspace methods for the reduction of very large models of mechanical structures, specifically for machine tools.

Vorwort

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter der Arbeitsgruppe Numerik am Institut *Computational Mathematics* der Technischen Universität Braunschweig.

Mein besonderer Dank gilt der Institutsleiterin Frau Prof. Dr. Heike Faßbender für ihre vielfältige und wohlwollende Unterstützung als Mentorin dieser Arbeit.

Mein herzlicher Dank gilt auch Herrn Prof. Dr. Michael Zäh und Herrn Prof. Dr. Kemnitz für das Interesse an meiner Arbeit.

Danken möchte ich auch der Deutschen Forschungs Gemeinschaft für die finanzielle Förderung des Projektes WAZorFEM aus dem wesentliche Ergebnisse in diese Arbeit eingeflossen sind.

In diesem Zusammenhang möchte ich mich herzlich bei Herrn Thomas Bonin vom Institut für Werkzeugmaschinen und Betriebswissenschaften der TU München für die konstruktive Zusammenarbeit während und nach der Projektphase bedanken. Der enge Kontakt zu diesem Institut war für mich sehr wertvoll.

Meine Zeit am Institut wurde auch wesentlich durch meine Kollegen, namentlich André Bodendieck, Juan Pablo Amorocho Duran, André Eppler, Peter Stange und Alexander Vendl, geprägt. Für die angenehme Arbeitsatmosphäre sowie die vielen fachlichen und nichtfachlichen Diskussionen möchte ich mich an dieser Stelle herzlich bedanken.

Braunschweig, im Juni 2011

Andreas Soppa

Inhaltsverzeichnis

Inhaltsverzeichnis	vii
Abbildungsverzeichnis	xiii
Tabellenverzeichnis	xvii
1 Einleitung	1
1.1 Stand der Technik und Literaturübersicht	3
1.1.1 Werkzeugmaschinen	4
1.1.2 Modellbildung und Simulation von Werkzeugmaschinen	7
1.1.3 Modellreduktion	9
1.2 Ziel und Vorgehen der Arbeit	11
1.2.1 Zielsetzung	11
1.2.2 Vorgehen der Arbeit	11
1.2.3 Aufbau der Arbeit	13
I Grundlagen	17
2 Hilfsmittel der linearen Algebra	19
2.1 Eigen- und Singulärwertzerlegung	19
2.2 Normen	22
2.3 Matrix-Zerlegungen	25
2.4 Krylov-Unterräume	26
2.5 Projektionen	27
3 Systemtheoretische Grundlagen	29
3.1 Darstellung von Systemen	29
3.2 Übertragungsverhalten	33
3.2.1 Übertragungsfunktion	33

3.2.2	Impulsantwort	37
3.2.3	Momente	38
3.2.3.1	Momente Systeme erster Ordnung	38
3.2.3.2	Momente Systeme zweiter Ordnung	41
3.2.3.3	Markov Parameter	44
3.3	Stabilität und Passivität	45
3.3.1	Stabilität	45
3.3.2	Passivität	46
3.4	Steuer- und Beobachtbarkeit	47
3.4.1	Steuerbarkeit	47
3.4.2	Beobachtbarkeit	48
3.4.3	Gramsche Matrizen	50
3.4.4	Hankel-Singulärwerte	52
3.5	Zeit- und Frequenzbereichsräume und Normen	53
3.5.1	Signalnormen	53
3.5.1.1	Normen und Räume des Zeitbereichs	53
3.5.1.2	Normen und Räume des Frequenzbereichs	54
3.5.2	Systemnormen	56
4	Grundlagen der Strukturdynamik	59
4.1	Strukturmodelle	60
4.2	Die Modaltransformation	61
4.3	Dämpfung	63
4.3.1	Modaler Dämpfungsansatz	64
4.3.2	Proportionaler Dämpfungsansatz	65
4.4	Übertragungsfunktionen von Strukturmodellen	65
5	Grundlagen der Modellreduktion	67
5.1	Problemstellung der Modellreduktion	67
5.1.1	Problemstellung	67
5.1.2	Anspruch an das Reduktionsverfahren	69
5.2	Modellreduktion durch Projektion	70
II	Werkzeugmaschinen	75
6	Aufbau und Funktionsweise von Werkzeugmaschinen	77
6.1	Aufbau und Funktionsweise	77

6.2	Vorschubantriebe	80
6.2.1	Grundaufbau	80
6.2.2	Elektrische Antriebe	80
6.2.3	Mechanische Übertragungssysteme	81
6.2.4	Linear-Direktantriebe	81
6.3	Antriebsregelung	82
6.4	Dynamisches Verhalten	84
7	Simulation von Werkzeugmaschinen	87
7.1	Regelungssimulation von Werkzeugmaschinen	88
7.1.1	Regelungstechnische Systeme	88
7.1.2	Grundstruktur des mechatronischen Systems	90
7.2	Modellbildung der Regelung und der elektrischen Antriebe	91
7.2.1	Modellbildung der Regelung	91
7.2.1.1	P- und PI-Regler	91
7.2.1.2	Führungsgrößenfilter	93
7.2.2	Modellbildung der elektrischen Antriebe	94
7.2.2.1	Servomotor	94
7.2.2.2	Transistorsteller	95
7.2.3	Gleichungssystem der Regelung und der elektrischen Antriebe	96
7.3	Modellbildung des mechanischen Systems	98
7.3.1	Modellierungsansätze	98
7.3.1.1	Starrkörpersysteme	99
7.3.1.2	Finite-Elemente-Methode	101
7.3.1.3	Elastische Mehrkörpersysteme	101
7.3.2	FEM-Modellierung des mechanischen Systems	102
7.3.2.1	Netzunabhängige Kopplung	103
7.3.2.2	Parametrische FEM-Modellbausteine	105
7.3.2.3	Datenmodell für parametrische FEM-Modellbausteine	106
7.3.3	Bewegungsgleichungen im Zustandsraum	108
7.4	Modellordnungsreduktion	110
7.5	Kopplung der Gleichungssysteme	112
III	Modellreduktion in der Simulation von Werkzeugmaschinen	115
8	Übersicht bekannter Reduktionsmethoden	117
8.1	Proper Orthogonal Decomposition	118

8.2	Balanciertes Abschneiden	120
8.3	Modale Reduktion	124
8.4	Krylov-Unterraum-Verfahren	125
8.4.1	Momentenabgleich	127
8.4.2	Expliziter Momentenabgleich	127
8.4.3	Krylov-Unterraum-Verfahren erster Ordnung	129
8.4.3.1	Verfahren mit $\mathbf{s}_0 = \mathbf{0}$	130
8.4.3.2	Verfahren mit $\mathbf{s}_0 \in \mathbb{C}$	131
8.4.3.3	Verfahren mit mehr als einer Entwicklungsstelle	133
8.4.3.4	Numerische Aspekte	135
8.4.3.5	Rationale Krylov-Verfahren	136
8.4.3.6	Erzielung reellwertiger Systeme	139
8.4.4	Verfahren für Systeme zweiter Ordnung	139
8.4.4.1	Verfahren mit $\mathbf{s}_0 = \mathbf{0}$	142
8.4.4.2	Verfahren mit $\mathbf{s}_i \in \mathbb{C}$	144
8.4.4.3	Verfahren mit mehr als einer Entwicklungsstelle	145
8.4.4.4	Numerische Aspekte	146
8.4.4.5	Erzielung reellwertiger Systeme	149
8.4.4.6	Ungedämpfte und proportional gedämpfte Systeme	151
8.4.5	Diskussion der Krylov-Unterraum-Verfahren	152
8.4.5.1	Passivitäts- und Stabilitätserhaltung	153
8.4.5.2	Die Wahl geeigneter Entwicklungsstellen	154
8.4.5.3	Die Wahl der geeigneten Stufen der verwendeten Krylov- Unterräume	155
8.4.5.4	Globale Fehlerschranke	155
8.4.5.5	Adaptive Methoden	155
8.4.5.6	Systeme mit vielen Ein- und Ausgängen	156
8.5	\mathcal{H}_2 -optimale Verfahren	156
8.5.1	IRKA	157
8.5.2	Tangentiale Hermite-Interpolationsverfahren	158
8.5.3	MIRIAM	159
9	Fehlerbestimmung und adaptive Methoden	161
9.1	Analytische Methoden	162
9.1.1	Lokale Fehlerbestimmung	162
9.1.1.1	Fehlerbestimmung der Übertragungsfunktion	162

9.1.1.2	Fehlerbestimmung des k -ten Momentes der Übertragungsfunktion	163
9.1.2	Globale Fehlerbestimmung	164
9.2	Heuristische Methoden	165
9.2.1	Lokale Fehlerschätzer	165
9.2.1.1	Heuristische Methoden	165
9.2.1.2	Fehlerschätzer nach Grimme	166
9.2.2	Ein globaler Fehlerschätzer	167
9.3	Adaptive Methoden	168
9.3.1	Adaptive Auswahl von Entwicklungsstellen	168
9.3.2	Adaptiver rationaler Arnoldi-Algorithmus (AORA)	170
10	Modellreduktion von Werkzeugmaschinen	175
10.1	Anwendungsbeispiele	176
10.1.1	Fehlerbestimmung reduzierter Systeme	177
10.1.1.1	Fehlerbestimmung im Frequenzbereich	177
10.1.1.2	Fehlerbestimmung im Zeitbereich	179
10.1.2	Verwendete Beispielmodelle	179
10.1.2.1	Erstes Beispielmodell	180
10.1.2.2	Zweites Beispielmodell	180
10.2	Spezielle Anforderungen an die Reduktionsverfahren	180
10.3	Verfahren basierend auf dem Block-Arnoldi-Algorithmus	184
10.3.1	Der Block-Arnoldi-Algorithmus	184
10.3.2	Ein modifizierter iterativer rationaler Krylov-Algorithmus (MIRKA)	186
10.3.2.1	Auswahlprozess der Entwicklungsstellen	188
10.3.2.2	Auswahl der Entwicklungsstellen zur Stabilisierung	189
10.3.2.3	Auswahl der Entwicklungsstellen des Verfahrens MIRKA	189
10.3.2.4	Konvergenzkriterium	192
10.3.3	Anwendung auf Systeme zweiter Ordnung	194
10.3.4	Diskussion der Approximationseigenschaften	196
10.3.5	Zusammenfassung der Resultate	201
10.4	Verfahren basierend auf dem globalen Arnoldi-Algorithmus	202
10.4.1	Der Globale Arnoldi-Algorithmus	202
10.4.1.1	Notationen	202
10.4.1.2	Der globale Arnoldi-Algorithmus	203
10.4.1.3	Eigenschaften des globalen Arnoldi-Algorithmus	204
10.4.2	Der iterative rationale globale Arnoldi-Algorithmus (IRGA)	206

10.4.2.1	Der globale Arnoldi-Algorithmus für Systeme erster Ordnung . .	206
10.4.2.2	Der globale Arnoldi-Algorithmus für Systeme zweiter Ordnung .	208
10.4.2.3	IRGA für Systeme zweiter Ordnung	209
10.4.3	Untersuchung der benötigten Berechnungszeit	209
10.4.4	Adaptive Modellreduktion	211
10.4.4.1	Adaptive Wahl der Stufe der Krylov-Unterräume für Systeme erster Ordnung	211
10.4.4.2	Adaptive Wahl der Stufe der Krylov-Unterräume für Systeme zweiter Ordnung	214
10.4.4.3	Der adaptive iterative rationale globale Arnoldi-Algorithmus für Systeme zweiter Ordnung (AIRGA_2o)	217
10.4.4.4	Der vollständige adaptive iterative rationale globale Arnoldi- Algorithmus für Systeme zweiter Ordnung (VAIRGA_2o)	218
10.4.4.5	Diskussion der Approximationseigenschaften	221
10.4.5	Zusammenfassung der Resultate	229
10.5	Automatische Modellreduktion	233
10.5.1	Abschätzung des Approximationsfehlers	234
10.5.2	Automatische Reduktionsverfahren	235
10.5.2.1	VAIRGA_MM	237
10.5.2.2	Weitere automatische Reduktionsverfahren	237
10.5.3	Diskussion der automatischen Reduktion	238
10.5.4	Zusammenfassung der Resultate	241
11	Modellreduzierte Simulation von Werkzeugmaschinen	243
11.1	Untersuchung von Lastfällen	243
11.2	Gekoppelte Simulation	247
12	Zusammenfassung und Ausblick	253
A	Anhang	257
A.1	Erstes Beispielmodell reduziert auf die Dimensionen 16, 24, 32 und 40	257
A.2	Zweites Beispielmodell reduziert auf die Dimensionen 32, 40, 48 und 56	265
	Literaturverzeichnis	273

Abbildungsverzeichnis

1.1	Strukturmechanische Berechnung und Regelungssimulation von Werkzeugmaschinen	6
2.1	Schematische Darstellung einer orthogonalen und einer schiefen Projektion	28
5.1	Einsatz der Modellreduktion bei der Simulation von Werkzeugmaschinen	68
5.2	Reduktion eines Systems erster Ordnung durch Projektion	71
5.3	Reduktion eines Systems zweiter Ordnung durch Projektion	73
6.1	Werkzeugmaschine mit serieller Achskinematik	78
6.2	Wirkprinzip von CNC-gesteuerten Werkzeugmaschinen	79
6.3	Aufbau eines CNC-gesteuerten elektromechanischen Vorschubantriebes	80
6.4	Funktionsprinzip eines Synchron-Linearmotors	82
6.5	Kaskadierte Lageregelung für Vorschubantriebe von Werkzeugmaschinen	83
7.1	Blockorientierte Modellierung	89
7.2	Mechatronisches Gesamtmodell	90
7.3	Vereinfachtes Modell einer kaskadierten Lageregelung	92
7.4	Blockschaltbild des PI-Reglers	92
7.5	Tiefpassfilter	93
7.6	Elektrische Ersatzschaltbilder des Synchron- und des Gleichstrommotors	95
7.7	Blockschaltbild eines Servomotors	96
7.8	Blockschaltbild des Transistorstellers mit PT1-Verhalten	96
7.9	Regelungs- und elektrotechnischer Teil eines mechatronischen Gesamtsystems . .	97
7.10	Reglermodell mit Schaltern zum De- und Aktivieren der Regelkreise sowie V_n und V_I zur Aufschaltung von Vorsteuergrößen	98
7.11	Blockschaltbild des regelungs- und des elektrotechnischen Gleichungssystems in Zustandsdarstellung	98
7.12	Modellierungsverfahren einer mechanischen Struktur	100
7.13	Netzunabhängige Kopplung von FE-Strukturen	104

7.14	Explizite und interpolierende Starrkörperelemente	105
7.15	Arbeit mit Modellbausteinen in Verbindung mit konventionellen FEM- Programmsystemen	107
7.16	Blockschaltbild der Bewegungsgleichungen in Zustandsdarstellung	110
7.17	Blockschaltbild des regelungs- und elektrotechnischen sowie des reduzierten me- chanischen Gleichungssystems im Zustandsraum	112
7.18	Zustandsraumdarstellung des geregelten mechatronischen Gesamtsystems	114
9.1	Schema zur Kategorisierung von Fehlerbestimmungsmethoden	161
10.1	FEM-Modell einer Werkzeugmaschine mit 4983 Freiheitsgraden - Erstes Bei- spielmodell	181
10.2	Frequenzgang des MIMO-Systems - Erstes Beispielmodell	182
10.3	FEM-Modell einer Werkzeugmaschine mit 54.885 Freiheitsgraden - Zweites Bei- spielmodell	183
10.4	Frequenzgang des MIMO-Systems - Zweites Beispielmodell	184
10.5	(Vereinfachtes) Ablaufdiagramm des Algorithmus <i>Stellenauswahl</i> zur Auswahl neuer Entwicklungsstellen	190
10.6	(Vereinfachtes) Ablaufdiagramm des <i>Modifizierten iterativen rationalen Krylov- Algorithmus</i> (MIRKA)	194
10.7	Relative Approximationsfehler der mit den Verfahren MODAL_2o, MIRIAM und MIRA_2o reduzierten Systeme - erstes Beispielmodell	199
10.8	Relative Approximationsfehler der mit den Verfahren MIRIAM und MIRA_2o reduzierten Systeme bis 10000 Hz. - erstes Beispielmodell	200
10.9	Benötigte Berechnungszeiten der Verfahren MIRA_2o und IRGA_2o - erstes Bei- spielmodell	211
10.10	Ablaufdiagramm des <i>Adaptiven iterativen rationalen globalen Arnoldi- Algorithmus für Systeme zweiter Ordnung</i> (AIRGA_2o)	219
10.11	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND - erstes Beispielmodell	223
10.12	Relative Fehler der reduzierten Systeme MODAL_2o, AIRGA_ND und VAIR- GA_ND - erstes Beispielmodell	224
10.13	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD - erstes Beispielmodell	225
10.14	Relative Fehler der reduzierten Systeme MODAL_2o, AIRGA_PD und VAIR- GA_PD - erstes Beispielmodell	226
10.15	Relative Approximationsfehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND - zweites Beispielmodell	229

10.16	Relative Fehler der reduzierten Systeme MODAL_2o, AIRGA_ND und VAIRGA_ND - erstes Beispielmodell	230
10.17	Relative Approximationsfehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD - zweites Beispielmodell	231
10.18	Relative Fehler der reduzierten Systeme MODAL_2o, AIRGA_PD und VAIRGA_PD - erstes Beispielmodell	232
10.19	Schema einer automatischen Reduktion mit Fehlerschätzer	233
10.20	Ablaufdiagramm des automatisierten Reduktionsverfahrens VAIRGA_2o	236
10.21	Gewählte Fehlerschranken ϵ_S und exakter Approximationsfehler des automatisch reduzierten Systems - erstes Beispielmodell	239
10.22	Gewählte Fehlerschranken ϵ_S und exakter Approximationsfehler des automatisch reduzierten Systems - zweites Beispielmodell	240
11.1	Verwendete Funktionen zur Simulation eines statischen bzw. eines dynamischen Lastfalles	244
11.2	Absoluter Approximationsfehler der Simulationsergebnisse eines statischen und eines dynamischen Lastfalles - erstes Beispielmodell	246
11.3	Absoluter Approximationsfehler der Simulationsergebnisse eines statischen und eines dynamischen Lastfalles - erstes Beispielmodell	247
11.4	Absoluter Approximationsfehler der Simulationsergebnisse eines statischen und eines dynamischen Lastfalles - erstes Beispielmodell	248
11.5	Oberste Hierarchieebene der kaskadierten Lageregelung ² in der Simulationsumgebung MATLAB/Simulink®	249
11.6	Absoluter Approximationsfehler $\delta_{abs}(t, 1, 4)$ des Simulationsergebnisses der verschiedenen reduzierten Systeme in Kopplung mit einem kaskadierten Regelkreis.	251
A.1	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND der Dimension 16 - erstes Beispielmodell	258
A.2	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD der Dimension 16 - erstes Beispielmodell	259
A.3	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND der Dimension 24 - erstes Beispielmodell	260
A.4	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD der Dimension 24 - erstes Beispielmodell	261
A.5	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND der Dimension 32 - erstes Beispielmodell	262
A.6	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD der Dimension 32 - erstes Beispielmodell	262

A.7	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIR- GA_ND und VAIRGA_ND der Dimension 40 - erstes Beispielmodell	263
A.8	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIR- GA_PD und VAIRGA_PD der Dimension 40 - erstes Beispielmodell	264
A.9	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIR- GA_ND und VAIRGA_ND der Dimension 32 - zweites Beispielmodell	266
A.10	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIR- GA_PD und VAIRGA_PD der Dimension 32 - zweites Beispielmodell	267
A.11	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIR- GA_ND und VAIRGA_ND der Dimension 40 - zweites Beispielmodell	268
A.12	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIR- GA_PD und VAIRGA_PD der Dimension 40 - zweites Beispielmodell	268
A.13	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIR- GA_ND und VAIRGA_ND der Dimension 48 - zweites Beispielmodell	269
A.14	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIR- GA_PD und VAIRGA_PD der Dimension 48 - zweites Beispielmodell	270
A.15	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIR- GA_ND und VAIRGA_ND der Dimension 56 - zweites Beispielmodell	271
A.16	Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIR- GA_PD und VAIRGA_PD der Dimension 56 - zweites Beispielmodell	271

Tabellenverzeichnis

7.1	Komponentenklassen für Gestell und Antriebsstrukturen sowie die entsprechenden Diskretisierungsansätze für deren Modellbausteine	106
10.1	Ergebnisse der verwendeten Verfahren	198
10.2	Aufwand des Block-Arnoldi bzw. des globalen Arnoldi-Algorithmus zur Berechnung von k Iterationen	206
10.3	Ergebnisse der untersuchten Reduktionsverfahren - erstes Beispielmodell	222
10.4	Entwicklungsstellen der letzten Iteration der untersuchten Reduktionsverfahren - erstes Beispielmodell	222
10.5	Reihenfolge der Verwendung der Entwicklungsstellen in der letzten Iteration der untersuchten Reduktionsverfahren - erstes Beispielmodell	223
10.6	Ergebnisse der untersuchten Reduktionsverfahren - zweites Beispielmodell	227
10.7	Entwicklungsstellen der letzten Iteration der untersuchten Reduktionsverfahren - zweites Beispielmodell	228
10.8	Reihenfolge der Verwendung der Entwicklungsstellen in der letzten Iteration der untersuchten Reduktionsverfahren - zweites Beispielmodell	228
10.9	Verwendete Fehlermaße zur Bestimmung der Entwicklungsstellen und zur Abschätzung des Approximationsfehlers des reduzierten Systems der beschriebenen Verfahren	238
10.10	Ergebnisse der verwendeten automatischen Verfahren - erstes Beispielmodell	239
10.11	Ergebnisse der verwendeten automatischen Verfahren - zweites Beispielmodell	240
A.1	Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 16	258
A.2	Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 24	260
A.3	Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 32	261
A.4	Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 40	263
A.5	Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 32	266
A.6	Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 40	267
A.7	Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 48	269

A.8	Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 56	270
-----	--	-----

Formelzeichen, Schreibweisen und Abkürzungen

Schreibweisen

Vektoren und Matrizen werden in dieser Arbeit durch fettgedruckte Klein- bzw. Großbuchstaben bezeichnet.

Formelzeichen

Mathematische Ausdrücke

\mathbb{R}	Körper der reellen Zahlen
\mathbb{C}	Körper der komplexen Zahlen
\mathbb{K}	Körper der reellen oder der komplexen Zahlen
$\mathcal{L}_p(\mathcal{I})$	Banachraum der auf dem Intervall \mathcal{I} p mal Lebesgue-integrierbaren Funktionen
\mathbb{Z}	Zustandsraum eines Systems
\mathbb{U}	Eingangsraum eines Systems
\mathbb{Y}	Ausgangsraum eines Systems
\mathbb{K}^n	Menge aller Vektoren der Dimension n mit Einträgen aus \mathbb{K}
$\mathbb{K}^{n \times m}$	Menge aller Matrizen der Dimension $n \times m$ mit Einträgen aus \mathbb{K}
$\mathcal{K}_k(\cdot, \cdot)$	Krylov-Unterraum der Stufe k
$\mathcal{G}_k(\cdot, \cdot, \cdot)$	Krylov-Unterraum zweiter Ordnung der Stufe k
\mathbf{V}, \mathbf{W}	Projektionsmatrizen
i	imaginäre Einheit ($\sqrt{-1}$)
$\operatorname{Re}(s)$	Realteil von s
$\operatorname{Im}(s)$	Imaginärteil von s

\mathbb{R}_+	Menge der positiven reellen Zahlen ($s \in \mathbb{R}_+ : s > 0$)
\mathbb{R}_-	Menge der negativen reellen Zahlen ($s \in \mathbb{R}_- : s < 0$)
\mathbb{C}_+	rechte Halbebene der komplexen Ebene ($s \in \mathbb{C}_+ : \operatorname{Re}(s) > 0$)
\mathbb{C}_-	linke Halbebene der komplexen Ebene ($s \in \mathbb{C}_- : \operatorname{Re}(s) < 0$)
\bar{s}	komplex-konjugierte von s
\mathbf{I}	Einheitsmatrix (mit geeigneter Dimension)
$\mathbf{0}$	Nullmatrix (mit geeigneter Dimension)
Σ	Diagonalmatrix mit Singulärwerten auf der Diagonalen
\mathbf{A}^T	Transponierte der Matrix \mathbf{A}
\mathbf{A}^*	komplex konjugiert Transponierte der Matrix \mathbf{A} , wenn $\mathbf{A} \in \mathbb{C}^{n \times m}$ Transponierte der Matrix \mathbf{A} , wenn $\mathbf{A} \in \mathbb{R}^{n \times m}$
$\mathbf{A} > 0$	\mathbf{A} ist positiv definit
$\mathbf{A} < 0$	\mathbf{A} ist negativ definit
$\mathbf{A} \geq 0$	\mathbf{A} ist positiv semidefinit
$\mathbf{A} \leq 0$	\mathbf{A} ist negativ semidefinit
$[\mathbf{A}]_{*i}$	i -te Spalte der Matrix \mathbf{A}
$[\mathbf{A}]_{i*}$	i -te Zeile der Matrix \mathbf{A}
$[\mathbf{A}]_{ij}$	Eintrag in der i -te Zeile und der j -ten Spalte der Matrix \mathbf{A}
$[\mathbf{v}]_i$	i -te Komponente eines Vektors \mathbf{v}
$(\mathbf{B})_{ij}$	Matrix in der i -te (Block-) Zeile und der j -ten (Block-) Spalte einer Block-Matrix \mathbf{B}
$\lambda_i(\mathbf{A})$	i -ter Eigenwert der Matrix \mathbf{A}
$\sigma_i(\mathbf{A})$	i -test größter Hankel-Singulärwert der Matrix \mathbf{A}
$\operatorname{span}\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$	Der durch die n Vektoren $\mathbf{v}_1, \dots, \mathbf{v}_n$ aufgespannte Raum
$\operatorname{colspan}\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$	Der durch die n Spaltenvektoren $\mathbf{v}_1, \dots, \mathbf{v}_n$ aufgespannte Raum
$\operatorname{colspan}(\mathbf{A})$	Der durch die Spaltenvektoren der Matrix \mathbf{A} aufgespannte Raum
$\mathcal{B}(\mathbf{A})$	Bild von \mathbf{A}
$\mathcal{N}(\mathbf{A})$	Kern von \mathbf{A}
$\mathcal{V} \oplus \mathcal{W}$	Direkte Summe der beiden Räume \mathcal{V} und \mathcal{W}
$\operatorname{diag}(a_1, \dots, a_n)$	Diagonalmatrix mit den Diagonaleinträgen a_1, \dots, a_n
$\operatorname{rang}(\mathbf{A})$	der Rang der Matrix \mathbf{A}
$\operatorname{rang}(\mathbf{A}, \tau)$	numerischer Rang der Matrix \mathbf{A} zum Parameterwert τ
$\det(\mathbf{A})$	die Determinante der Matrix \mathbf{A}
$\operatorname{spur}(\mathbf{A})$	Spur der Matrix \mathbf{A}

$\mathbf{A} \otimes \mathbf{B}$	Kronecker-Produkt der Matrizen \mathbf{A} und \mathbf{B}
$ s $	Betrag von s
$\ \mathbf{x}\ _p$	p -Vektornorm von \mathbf{x}
$\ \mathbf{x}\ $	2-Vektornorm von \mathbf{x}
$\ \mathbf{A}\ _p$	durch die p -Vektornorm induzierte Matrixnorm von \mathbf{A}
$\ \mathbf{A}\ $	durch die 2-Vektornorm induzierte Matrixnorm von \mathbf{A}
$\ \mathbf{A}\ _F$	Frobenius-Norm von \mathbf{A}
$\ \mathbf{H}(s)\ _{\mathcal{H}_p}$	\mathcal{H}_p -Norm einer matrixwertigen Funktion $\mathbf{H}(s)$
$\ \mathbf{H}(s)\ _{\mathcal{H}_2}$	\mathcal{H}_2 -Norm einer matrixwertigen Funktion $\mathbf{H}(s)$
$\ \Sigma\ _H$	Hankel-Norm eines Systems Σ

Bezeichnungen verwendeter Systeme

Σ_{2o}	Kurzschreibweise eines Systems zweiter Ordnung
Σ_{1o}	Kurzschreibweise eines Systems erster Ordnung
$\mathbf{M}, \mathbf{D}, \mathbf{K}, \mathbf{f}, \mathbf{c}_p^T, \mathbf{c}_v^T$	Matrizen eines SISO-Systems zweiter Ordnung
$\mathbf{M}, \mathbf{D}, \mathbf{K}, \mathbf{F}, \mathbf{C}_p, \mathbf{C}_v$	Matrizen eines MIMO-Systems zweiter Ordnung
$\hat{\mathbf{M}}, \hat{\mathbf{D}}, \hat{\mathbf{K}}, \hat{\mathbf{f}}, \hat{\mathbf{c}}_p^T, \hat{\mathbf{c}}_v^T$	reduzierte Matrizen eines MIMO-Systems zweiter Ordnung
$\hat{\mathbf{M}}, \hat{\mathbf{D}}, \hat{\mathbf{K}}, \hat{\mathbf{F}}, \hat{\mathbf{C}}_p, \hat{\mathbf{C}}_v$	reduzierte Matrizen eines MIMO-Systems zweiter Ordnung
$\mathbf{E}, \mathbf{A}, \mathbf{b}, \mathbf{c}^T$	Matrizen eines SISO-Systems erster Ordnung
$\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}$	Matrizen eines MIMO-Systems erster Ordnung
$\hat{\mathbf{E}}, \hat{\mathbf{A}}, \hat{\mathbf{b}}, \hat{\mathbf{c}}^T$	reduzierte Matrizen eines SISO-Systems erster Ordnung
$\hat{\mathbf{E}}, \hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}$	reduzierte Matrizen eines MIMO-Systems erster Ordnung
n	Dimension des Systems zweiter Ordnung
r	Dimension des reduzierten Systems
m	Anzahl der Eingänge eines Systems
\tilde{m}	Anzahl der linear unabhängigen Eingänge eines Systems
q	Anzahl der Ausgänge eines Systems
\mathbf{z}	Zustandsvektor eines Systems erster Ordnung
\mathbf{x}	Zustandsvektor eines Systems zweiter Ordnung
$\hat{\mathbf{z}}$	Zustandsvektor eines reduzierten Systems erster Ordnung
$\hat{\mathbf{x}}$	Zustandsvektor eines reduzierten Systems zweiter Ordnung
u	Eingangsfunktion eines SISO-Systems
\mathbf{u}	Vektor der Eingangsfunktionen eines MIMO-Systems

y	Ausgangsfunktion eines SISO-Systems
\mathbf{y}	Vektor der Ausgangsfunktionen eines MIMO-Systems
\hat{y}	Ausgangsfunktion eines reduzierten SISO-Systems
$\hat{\mathbf{y}}$	Ausgangsfunktionenvektor eines reduzierten MIMO-Systems
F_S	Vektor der äußeren Störkräfte
$\mathbf{u}_A, \mathbf{u}_S$	Separierte Eingangsvektoren eines mechanischen Systems
$\mathbf{y}_M, \mathbf{y}_S$	Separierte Ausgangsvektoren eines mechanischen Systems
$\mathbf{F}_A, \mathbf{F}_S$	Eingriffsmatrizen eines mechanischen Systems
$\mathbf{B}_A, \mathbf{B}_S$	Separierte Eingangsmatrizen eines mechanischen Systems
$\mathbf{C}_{p,M}, \mathbf{B}_{v,M}$	Messgrößenausgänge eines mechanischen Systems
$\mathbf{C}_{p,S}, \mathbf{B}_{v,S}$	Störstellenausgänge eines mechanischen Systems
$\mathbf{E}_R, \mathbf{A}_R, \mathbf{b}_R, \mathbf{c}_R^T$	Matrizen eines regelungs-/elektrotechnischen SISO-Systems
$\mathbf{E}_R, \mathbf{A}_R, \mathbf{B}_R, \mathbf{C}_R$	Matrizen eines regelungs-/elektrotechnischen MIMO-Systems
$\mathbf{B}_{R,s}, \mathbf{B}_{R,i}$	Separierte Eingangsmatrizen eines regelungs-/elektrotechnischen Systems
\mathbf{z}_R	Zustandsvektor eines regelungs-/elektrotechnischen Systems
\mathbf{z}_{DR}	Zustände der Drehzahlregler eines regelungs-/elektrotechnischen Systems
\mathbf{z}_{SR}	Zustände der Stromahlregler eines regelungs-/elektrotechnischen Systems
\mathbf{z}_{TR}	Zustände der Transistorsteller eines regelungs-/elektrotechnischen Systems
\mathbf{z}_{SM}	Zustände der Servomotoren eines regelungs-/elektrotechnischen Systems
$\mathbf{u}_{R,s}, \mathbf{u}_{R,i}$	Separierte Eingangsvektoren eines regelungs-/elektrotechnischen Systems
\mathbf{y}_R	Ausgangsfunktion eines regelungs-/elektrotechnischen Systems
$\mathbf{E}_G, \mathbf{A}_G, \mathbf{b}_G, \mathbf{c}_G^T$	Matrizen eines mechatronischen SISO-Gesamtsystems
$\mathbf{E}_G, \mathbf{A}_G, \mathbf{B}_G, \mathbf{C}_G$	Matrizen eines mechatronischen MIMO-Gesamtsystems
$\mathbf{E}_A, \mathbf{E}_S$	Eingriffsmatrizen zur Einleitung von Kräften und Momenten ins mechanische System
\mathbf{z}_G	Zustandsvektor eines mechatronischen MIMO-Gesamtsystems
\mathbf{u}_G	Eingangsvektor eines mechatronischen MIMO-Gesamtsystems
\mathbf{y}_G	Ausgangsvektor eines mechatronischen MIMO-Gesamtsystems
h_j	j -tes Moment der Übertragungsfunktion eines SISO-Systems

\mathbf{H}_j	j -tes Moment der Übertragungsfunktion eines MIMO-Systems
$\angle(\mathbf{H}(s))$	Winkel von $\mathbf{H}(s)$ <i>Mechatronik</i>
I	Stromstärke
U	Spannung
R	Ohmscher Widerstand
L	Induktivität
M_M	Motormoment
K_M	Drehmomentkonstante eines Servomotors
n	Drehmoment eines Motors
<i>Regelungstechnik</i>	
K_P, K_A, T_N	Proportionalbeiwert, Anpassungsfaktor und Nachstellzeit eines Reglers
K_V	Proportionalbeiwert eines Lagereglers
T_R, f_w, p_T	Rechenzeit, Schaltfrequenz und Rechenzeit eines Transistorstellers
T_T	Totzeit eines Transistorstellers

Abkürzungen

API	Englisch: <u>A</u> pplication <u>P</u> rogramming <u>I</u> nterface
BTA	Approximation durch balanciertes Abschneiden (In Englisch: <u>B</u> alanced <u>T</u> runcation <u>A</u> pproximation)
CAD	Englisch: <u>C</u> omputer- <u>A</u> ided <u>D</u> esign
CNC	Englisch: <u>C</u> omputerized <u>N</u> umerical <u>C</u> ontrol
DGL	<u>D</u> ifferenzial <u>g</u> leichung
DAE	Algebro-Differenzialgleichung (Englisch: <u>D</u> ifferential <u>A</u> lgebraic <u>E</u> quation)
FE	<u>F</u> inite- <u>E</u> lemente
FEM	<u>F</u> inite- <u>E</u> lemente- <u>M</u> ethode
LTI	linear zeitinvariant (In Englisch: <u>L</u> inear <u>T</u> ime <u>I</u> nvariant)
MIMO	<u>M</u> ultiple <u>I</u> nput <u>M</u> ultiple <u>O</u> utput
MPC	Englisch: <u>M</u> ultipoint <u>C</u> onstraints

RBE	Englisch: <u>R</u> igid <u>B</u> ody <u>E</u> lement
SISO	<u>S</u> ingle <u>I</u> nput <u>S</u> ingle <u>O</u> utput
TCP	Englisch: <u>T</u> ool <u>C</u> enter <u>P</u> oint
UML	Englisch: <u>U</u> nified <u>M</u> odelling <u>L</u> anguage
VLSI	<u>V</u> ery- <u>L</u> arge- <u>S</u> cale <u>I</u> ntegration

Abkürzungen für Krylov-Unterraum-Verfahren

SOAR	Englisch: <u>S</u> econd <u>O</u> der <u>A</u> rnoldi Algorithm
IRKA	iterativer rationaler Krylov-Algorithmus (In Englisch: <u>I</u> terative <u>R</u> ational <u>K</u> rylov <u>A</u> lgorithm)
MIRIAM	MIMO iterativer rationaler Interpolations-Algorithmus (In Englisch: <u>M</u> IMO <u>I</u> terative <u>R</u> ational <u>I</u> nterpolation <u>A</u> lgorithm)
AORA	adaptiver rationaler Krylov-Algorithmus (In Englisch: <u>A</u> daptive <u>O</u> der <u>R</u> ational <u>A</u> rnoldi <u>A</u> lgorithm)
AORGA	adaptiver rationaler globaler Arnoldi-Algorithmus (In Englisch: <u>A</u> daptive <u>O</u> der <u>R</u> ational <u>G</u> lobal <u>A</u> rnoldi Algorithm)
MIRKA	modifizierter iterativer rationaler Krylov-Algorithmus (In Englisch: <u>M</u> odified <u>I</u> terative <u>R</u> ational <u>K</u> rylov <u>A</u> lgorithm)
MIRL	modifizierter iterativer rationaler Lanczos-Algorithmus (In Englisch: <u>M</u> odified <u>I</u> terative <u>R</u> ational <u>L</u> anczos <u>A</u> lgorithm)
MIRA	modifizierter iterativer rationaler Arnoldi-Algorithmus (In Englisch: <u>M</u> odified <u>I</u> terative <u>R</u> ational <u>A</u> rnoldi <u>A</u> lgorithm)
IRGA	iterativer rationaler globaler Arnoldi-Algorithmus (In Englisch: <u>I</u> terative <u>R</u> ational <u>G</u> lobal <u>A</u> rnoldi <u>A</u> lgorithm)
AIRGA	adaptiver iterativer rationaler globaler Arnoldi-Algorithmus (In Englisch: <u>A</u> daptive <u>I</u> terative <u>R</u> ational <u>G</u> lobal <u>A</u> rnoldi <u>A</u> lgorithm)
VAIRGA	vollständig adaptiver iterativer rationaler globaler <u>A</u> rnoldi-Algorithmus)

Kapitel 1

Einleitung

Die Herstellung von Werkzeugmaschinen steht durch den wachsenden weltweiten Konkurrenzdruck vor großen Herausforderungen. In den letzten Jahren entwickelte sich der Einsatz und die Produktion von Werkzeugmaschinen in Asien mit einer deutlich höheren Dynamik als in Europa und Amerika [99]. Angetrieben wurde diese Entwicklung durch die Vorteile der Wettbewerber aus Schwellenländern durch deren günstige Lohnkostenstruktur. Dabei wurde der Gewinn an Marktanteilen vorwiegend durch den Produktionszuwachs im wachstumsstarken Niedrigpreissegment erzielt [2]. Europäische und amerikanische Anbieter bedienen dagegen vorwiegend die Nachfrage nach hochentwickelter Produktionstechnik. Nach einer Sättigung des Marktes im Niedrigpreissegment ist in zunehmenden Maße ein Gewinn an Marktanteilen asiatischer Wettbewerber auch in anderen Preissegmenten zu beobachten. Damit werden diese Anbieter zu Konkurrenten auf dem Markt der hochentwickelten Produktionstechnik. Um ihre Marktanteile in diesem Segment aufrecht zu erhalten, sind die Hersteller von Werkzeugmaschinen gezwungen in immer kürzeren Zyklen neue Produkte, die sich durch einen erhöhten Anwendernutzen und geringeren Herstellungskosten auszeichnen, auf den Markt zu bringen.

Ein erhöhter Nutzen für den Anwender ist dabei in einer höheren Produktivität und Flexibilität der am Markt angebotenen Werkzeugmaschinen zu sehen. Um Anwendern von Werkzeugmaschinen Produkte mit diesen erhöhten Anwendernutzen bieten zu können, werden unterschiedliche Ansätze verfolgt. In den vergangenen Jahren gehörten beispielsweise die informationstechnische Vernetzung von Maschinen, die Entwicklung neuer Werkstoffe und aktiver Strukturen, die Dezentralisierung von Maschinenfunktionen oder die Digitalisierung von Maschinenfunktionen [186] dazu. Dabei gewinnt bei der Funktionserweiterung und der Leistungssteigerung neuer Werkzeugmaschinen die Automatisierungstechnik zunehmend an Bedeutung.

Die wichtigsten Beurteilungskriterien für Werkzeugmaschinen sind das dynamische Verhalten und die Genauigkeit des Bearbeitungsprozesses. Die Prozesstechnik hat durch die Entwicklung neuer Technologien zu hohen Leistungssteigerungen geführt. Die benötigten Beiträge zur Steigerung der Vorschubgeschwindigkeiten hat dabei die Antriebstechnik geliefert. Dies ist darauf zurück zu führen, dass im Bereich der Antriebstechnik heute leistungsstarke Servoantriebe zur Verfügung stehen, die durch digitale Steuerungs- und Regelungsmodule über weite Drehzahlbereiche regelbar sind. Zudem sind zur dynamisch genauen Positionsregelung hochauflösende

optische Messgeräte verwendbar, die eine schnelle und exakte Positions- und Winkelbestimmung gewährleisten [138].

Die Grenzen bei der Dynamik und der Genauigkeit von Werkzeugmaschinen ergeben sich aus regelungstechnischer Sicht durch die Schwingungen der Maschinenstruktur. Durch die Signale der Positionsmessgeräte und der Drehgeber in die Antriebsregelkreise werden diese rückgekoppelt, was bei zu hohen Reglerverstärkungen zu Instabilitäten führen kann [138]. Hierin liegt die Ursache dafür, dass die Antriebsregelungen nur mit begrenzten Bandbreiten betrieben werden können [1].

Durch Erweitern der Regelungsalgorithmen, indem das Verhalten der Regelung z.B. mit Filtern oder aktiver Dämpfung optimal an die Eigenschaften der Regelstrecke angepasst wird, lässt sich die dynamische Genauigkeit einer Werkzeugmaschine verbessern. Um eine maximale Leistungssteigerung zu erreichen, ist jedoch die optimale Auslegung der mechanischen Konstruktion im Hinblick auf ihr Zusammenspiel mit der Antriebsregelung und dem Bearbeitungsprozess entscheidend. Diese Auslegungsaufgaben stellen die Produktentwicklung vor große Aufgaben, da es sich bei Werkzeugmaschinen um komplexe mechatronische Systeme handelt, die aus mechanischen, elektrotechnischen und informationstechnischen Komponenten bestehen [138].

Zur Überprüfung und Absicherung der Systemeigenschaften sind bei einer konventionellen Vorgehensweise aufwendige Experimente und Versuchsreihen notwendig. Der dadurch entstehende finanzielle und zeitliche Aufwand kann jedoch durch den Einsatz rechnergestützter Methoden deutlich reduziert werden. Ein hilfreiches Mittel zur Unterstützung des Produktentwicklungsprozesses ist in der Computersimulation zu sehen, da Maschinenentwürfe damit vor einer Investition in Prototypen analysiert und optimiert werden können.

Durch die Leistungssteigerungen in den letzten Jahren stehen heute im Bereich der Hard- und Software numerische Berechnungswerkzeuge zur Verfügung, mit deren Einsatz sich die Simulation auch sehr komplexer technischer Systeme anhand von digitalen Modellen effizient und kostengünstig durchführen lassen. Deshalb stellen Computersimulationen ein wichtiges Instrument dar, um das Verhalten von Werkzeugmaschinen möglichst früh während des Entwicklungsprozesses analysieren zu können [138].

Da Werkzeugmaschinen heutzutage komplexe mechatronische Gebilde sind, die als System betrachtet werden müssen, gewinnt die gesamtheitliche Simulation von Komponenten verschiedener Fachdisziplinen dabei zunehmend an Bedeutung. So fließt in eine gesamtheitliche Simulation neben der Beschreibung der mechanischen Struktur auch die Beschreibung der Antriebstechnik, der Hydraulik oder der Regelungstechnik mit ein [138].

Strukturmechanische Berechnungen mit Hilfe der Finite-Elemente-Methode (FEM) sind bereits ein fester Bestandteil des Produktionsentwicklungsprozesses geworden. Dabei wird diese Methode vorwiegend zur Auslegung der Gestellstrukturen durchgeführt. Eine Simulation und regelungstechnische Analyse erfolgt in der Regel in getrennten Betrachtungen. In [138] wird ein Verfahren zur strukturmechanischen Berechnung und Regelsimulation von Werkzeugmaschinen auf einem Computermodell des Maschinengestells, der Antriebsstränge, der elektrischen Antriebe und der Antriebsregelung vorgeschlagen. Dabei erfolgt die Modellbildung des Maschinengestells und der Antriebssysteme anhand parametrischer Modellbausteine.

Da die Modellbildung mit der FEM bei einem Strukturmodell mit einem hohen Detaillierungsgrad zu sehr hochdimensionalen Gleichungssystemen führt, ist eine Simulation mit diesen Systemen praktisch nicht durchführbar. Aus diesem Grund wird vor der Zusammenführung des elektro- und des informationstechnischen Modells eine Modellreduktion des mechanischen Modells durchgeführt.

In dieser Arbeit werden angepasste mathematische Verfahren zur automatisierten Modellreduktion der Finite-Elemente-Modelle, die durch die Modellierung rotatorischer Antriebssysteme mit Kugelgewindetrieb im Verbund mit dem Maschinengestell resultieren, entwickelt und auf die Prozesstauglichkeit hinsichtlich der modellreduzierten Simulation von Werkzeugmaschinen validiert.

Bei den in dieser Arbeit betrachteten Reduktionsverfahren handelt es sich um Projektionsverfahren mit Moment-Matching-Eigenschaft, die auf der Verwendung von Krylov-Unterräumen basieren. Das Prinzip des Moment-Matching basiert auf der Interpolation der Übertragungsfunktion des Modells mit Hilfe eines Abgleichs der Koeffizienten der Potenzreihenentwicklung der Übertragungsfunktion um eine oder mehrere Entwicklungsstellen in der komplexen Zahlenebene. Um diese Moment-Matching-Eigenschaften zu erfüllen, werden bei Krylov-Unterraum-Verfahren an den Entwicklungsstellen Basen von geeigneten Krylov-Unterräumen berechnet. Die in dieser Arbeit entwickelten Verfahren verwenden dazu die globale Arnoldi-Methode, die sich durch hohe Effizienz bei der Berechnung der Reduktion von Systemen mit mehreren Ein- und Ausgängen auszeichnet.

Des weiteren wird in dieser Arbeit eine mögliche Automatisierung des Reduktionsprozesses vorgestellt. Für Krylov-Unterraum-Verfahren existieren zwar Ansätze zur Wahl der Entwicklungsstellen und zur adaptiven Wahl der Stufe des zu jeder Entwicklungsstelle generierten Krylov-Unterraumes, jedoch existieren nach Kenntnis des Autors noch keine Verfahren, die die Wahl der Entwicklungsstellen mit einer adaptiven Bestimmung der Stufe des zugehörigen Krylov-Unterraumes kombinieren. In dieser Arbeit wird deshalb durch Modifikationen und Erweiterungen bestehender adaptiver Methoden ein Verfahren vorgestellt, das eine Automatisierung des Reduktionsprozesses ermöglicht.

Im Hinblick darauf wird im folgenden Abschnitt zunächst der derzeitige Stand der Wissenschaft und der Technik beleuchtet und das Ziel sowie der weitere Aufbau der vorliegenden Arbeit erläutert.

1.1 Stand der Technik und Literaturübersicht

In der vorliegenden Arbeit können mit dem Aufbau von Werkzeugmaschinen, der effizienten Simulation von mechatronischen Systemen und der linearen Modellreduktion drei Themengebiete identifiziert werden. Das erste Themengebiet umfaßt den Aufbau, die Funktionsweise und den Entwurfsprozess von Werkzeugmaschinen, das zweite Themengebiet deckt die effiziente Modellbildung und die Simulation von Werkzeugmaschinen ab und das dritte Themengebiet befaßt sich mit Themen der effizienten Reduktion von FEM-Modellen. Verbunden werden diese Themengebiete durch die Notwendigkeit modellierte FEM-Modelle in der Dimension zu reduzieren,

um Werkzeugsimulationen effizient durchführen zu können und damit den Entwurfprozess von Werkzeugmaschinen zu optimieren.

Der Fokus der vorliegenden Arbeit liegt auf der automatisierten und effizienten Reduktion von FEM-Modellen von Werkzeugmaschinen, wohingegen der Entwurfsprozess und die Simulation von Werkzeugmaschinen hier nur in ihren Grundzügen dargestellt werden können. Um den Entwicklungsprozess von Werkzeugmaschinen gesamtheitlich darzustellen, werden hier deshalb nur die Grundlagen des Aufbaus und der Simulation von Werkzeugmaschinen dargestellt. Diese orientieren sich dabei auf den Darstellungen in [138].

Zu allen drei Themengebieten existieren zahlreiche Veröffentlichungen, auf die in dieser Arbeit Bezug genommen wird. Dabei sollen diese Arbeit und die zitierten Veröffentlichungen eine Übersicht der Verzahnung der Themengebiete zur Zielerreichung einer effizienten Simulation von Werkzeugmaschinen geben. In den folgenden Abschnitten erfolgt eine Darstellung des derzeitigen Standes der Technik und der Wissenschaft, mit den Angaben der, im Bezug zum jeweiligen Themengebiet, wichtigsten Referenzen. Ergänzt wird die Übersicht durch eine ausführliche Darstellung der Grundlagen der verschiedenen Themengebiete in den Kapiteln 4 bis 7 und zahlreiche zusätzliche Literaturhinweise.

1.1.1 Werkzeugmaschinen

Werkzeugmaschinen in heutiger Erscheinungsform sind mechatronische Systeme. Unter einem mechatronischen System wird hier ein System verstanden, das aus einem mechanischen Teil (dem Maschinengestell) und einem regelungstechnischen Teil (elektronische Antriebe und Regelung) besteht. Zur Darstellung der Grundlagen des Aufbaus von Werkzeugmaschinen werden in dieser Arbeit Maschinen mit serieller Achskinematik betrachtet. Bei diesen Werkzeugmaschinen kann das Werkzeug bzw. das Werkstück je nach Kombination translatorischer und rotatorischer Vorschubachsen auf räumlichen Bahnkurven relativ zueinander bewegt werden. Die Beschreibung dieser Werkzeugmaschinen orientiert sich dabei an [138]. Die Achsbewegungen werden unter den Vorgaben einer numerischen Steuerung (CNC)¹ durch positionsgeregelte elektromechanische Vorschubantriebe, die aus einem elektrischen Antrieb und einem mechanischem Übertragungssystem bestehen, erzeugt. Die von der CNC erzeugten Führungsgrößen werden mit Hilfe einer informatonstechnisch verzweigten Übertragungsstrecke mit regelungstechnischen, elektrischen und mechanischen Teilstrecken auf die Prozessebene (Bearbeitungsstelle des Werkstücks) geleitet. Durch Rückführungen von Messgrößen liegt ein mehrfach rückgekoppeltes System vor, das ein komplexes Verhalten vorweist.

Als Antriebe elektromechanischer Vorschubantriebe werden heute vorwiegend permanenterregte Servo-Drehstrommotoren verwendet, die sich durch eine einfache Regelbarkeit und einer hohen Dynamik auszeichnen, siehe [29, 120, 159]. Die Versorgung der elektrischen Antriebe mit der benötigten elektrischen Energie erfolgt über Transistorsteller. Diese beziehen die benötigte elektrische Energie aus dem dreiphasigen Versorgungsnetz und stellen sie den Antrieben in der benötigten Form zur Verfügung.

¹Englisch: Computerized Numerical Control

Das heute bei Werkzeugmaschinen am weitesten verbreitete Wirkprinzip zur Realisierung linearer Vorschubbewegungen ist der Kugelgewindetrieb [138]. Als Alternative zum Kugelgewindetrieb wird bei Großwerkzeugen das Zahnstange-Ritzel-System verwendet [30].

Als leistungsfähige Alternative zu elektromechanischen Vorschubsystemen hat sich in den letzten Jahren der Linear-Direktantrieb entwickelt. Bei dieser Antriebsart erfolgt eine direkte Kraftübertragung am Vorschubschlitten, wodurch kein mechanisches Übertragungssystem benötigt wird und somit störende Elastizitäten bei der Kraftübertragung vom Motor auf den Vorschubschlitten entfallen. Als Nachteil dieser Antriebsart ist der niedrige Wirkungsgrad zu sehen, der hohe Verlustleistungen der Maschinenstruktur als auch der Motorwicklungen verursacht.

Heute verbreitete Systeme zur Positionsregelung der elektrischen Vorschubantriebe von Werkzeugmaschinen verwenden überwiegend eine kaskadierte Regelkreisstruktur [1, 160]. Bei dieser Regelkreisstruktur wird der auf der obersten Kaskadenebene angesiedelte Lageregler durch zwei Hilfsregelkreisen zur Drehzahl- und Stromregelung unterstützt.

Durch schnelle Positioniervorgänge mit hohen Beschleunigungen und hohem Ruck ergeben sich hohe statische und dynamische Belastungen für Werkzeugmaschinen. Die Bearbeitung eines Werkstücks mit hohen Leistungen erfordert deshalb eine steife Maschinenkonstruktion, um genaue und möglichst ebene Oberflächen zu erzeugen. Treten bei Bearbeitungsprozessen Schwingungen des Werkzeugs auf, wird eine wellige Oberfläche des Werkstücks erzeugt. Die Rückkopplung modulierter Kraftanteile auf die Maschine kann zur Entdämpfung der Schwingungen bis hin zur Instabilität führen, was sich akustisch durch Rattern der Maschine und am Werkstück durch Rattermarken bemerkbar macht [138].

Um die Positioniervorgänge zu beschleunigen, ist es nötig die Beschleunigung der Massen der Maschine zu erhöhen, die bewegt werden müssen. Um dieses Ziel zu erreichen, muss die bewegte Masse reduziert werden. Mit der weiterhin bestehenden Steifigkeitsanforderung an die Massen ergibt sich ein Problem, das nur durch Kompromisse gelöst werden kann. Optimale Lösungen dieses Problems ergeben sich durch die Verwendung von Leichtbaukonstruktionen. Wegen der bei Leichtbaukonstruktionen zunehmend zu beobachtenden strukturdynamischen Wechselwirkungen der Gestell- und Antriebsstrukturen müssen die Konstruktionen jedoch hinsichtlich ihrer dynamischen Übertragungseigenschaften im Regelkreis optimiert werden [138].

Aufgrund der gegenseitigen Abhängigkeit mechanischer, elektro- und informationstechnischer Funktionsträger in Aufbau, Funktionsweise und Verhalten stellen diese Auslegungsaufgaben den Entwurfprozess von Werkzeugmaschinen vor große Aufgaben [138]. Zur Unterstützung dieses Auslegungprozesses ist eine gesamtheitliche Simulation der Werkzeugmaschine als mechatronisches System erforderlich, dessen Eigenschaften vom Zusammenspiel der Maschinenstruktur und den Einflüssen der Regelung [26, 138] sowie des Bearbeitungsprozesses [182] bestimmt werden. Aus diesem Grund rückt die integrierte Simulation des Gesamtsystems, d.h. die Abbildung von Maschinenstruktur, Regelung und Prozess in einem Modell immer stärker in den Mittelpunkt. Die Abbildung 1.1 gibt den Ablauf der Modellierungs- und Berechnungsschritte für die strukturmekanischen Analysen und Regelungssimulationen im Entwurfsprozess von Werkzeugmaschinen wieder. Ein umfassender Überblick zur integrierten Modellbildung der Maschinenstruktur,

der Regelung und des Prozesses von Werkzeugmaschinen wird in [8] gegeben. Dabei werden sowohl die Finite-Elemente-Methode (FEM) als auch Mehrkörpersysteme (MKS) für strukturelle und kinematische Untersuchungen betrachtet. In der ersten Phase des Entwurfsprozesses

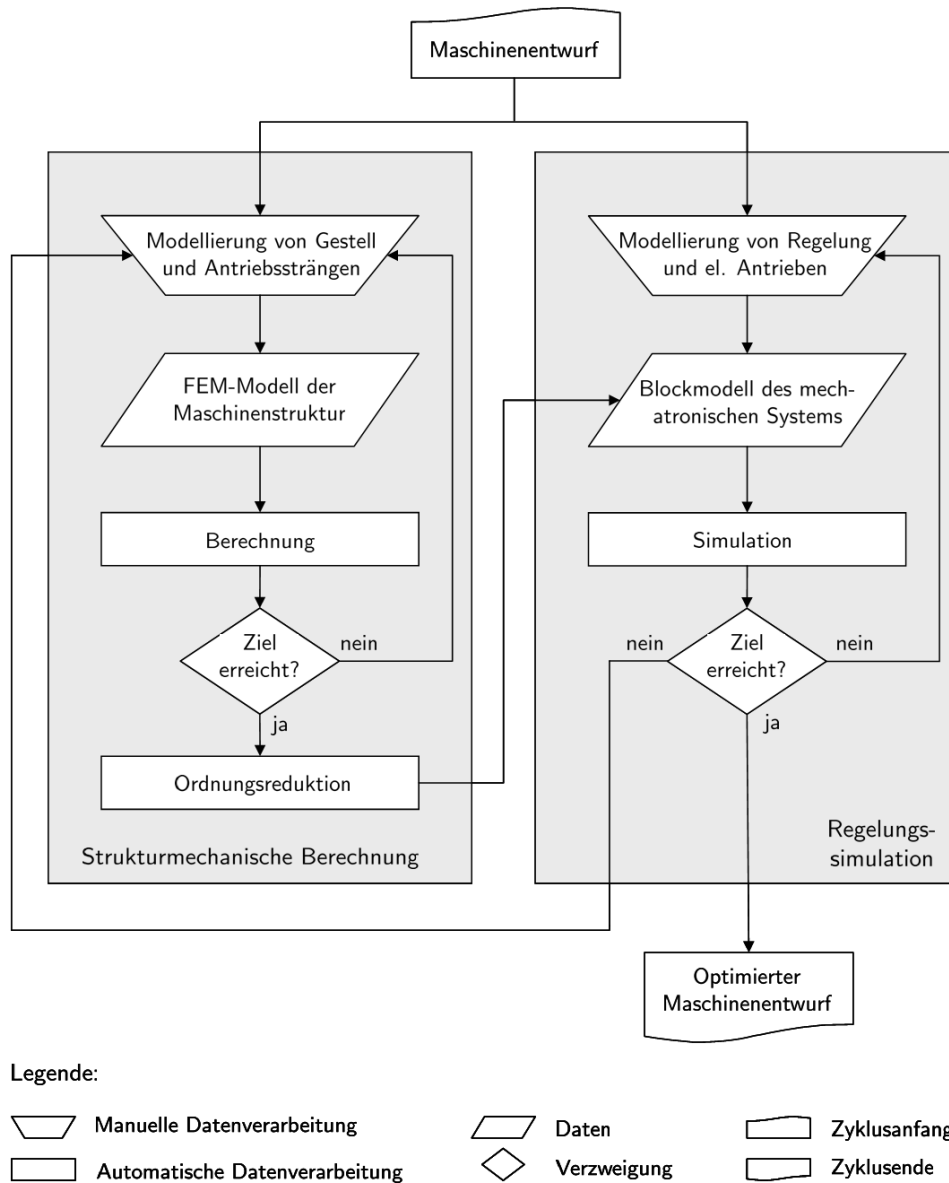


Abbildung 1.1: Strukturmechanische Berechnung und Regelungssimulation von Werkzeugmaschinen nach [138]

kann mit dem integrierten (und ggf. reduzierten) FEM-Modell der Maschine die mechanische Konstruktion nach rein strukturmechanischen Kriterien optimiert werden. Durch die Integration des FEM-Modells in das Blockmodell des mechatronischen Gesamtsystems über den Weg der Modellreduktion werden die Regelstrecken der Antriebsregelkreise beschrieben. Mit Hilfe der Regelungssimulation können nun die Parameter der Regelung optimiert werden, wobei die bandbreitenbegrenzenden Eigenfrequenzen sichtbar werden und Aufschluss über konstruktive Ursachen geben [138].

1.1.2 Modellbildung und Simulation von Werkzeugmaschinen

Die Genauigkeit des dynamischen Verhaltens von Werkzeugmaschinen lässt sich durch erweiterte Regelalgorithmen verbessern. Bei dieser Vorgehensweise sind der Steigerung der Genauigkeit allerdings Grenzen gesetzt. Soll das maximale Leistungspotential einer Werkzeugmaschine ausgeschöpft werden, so ist die optimale Auslegung der mechanischen Konstruktion im Hinblick auf ihr Zusammenspiel mit der Antriebsregelung entscheidend. Betrachtet man ein mechatronisches System in Bezug auf seinen Aufbau und seine Funktionsweise, kann es als regelungstechnisches System betrachtet werden. Bei Werkzeugmaschinen ergibt sich dabei eine Aufteilung des Regelkreises in ein informationstechnisches, ein elektrotechnisches und ein mechanisches Teilsystem.

Die Modellbildung des informations- und elektrotechnischen Teils erfolgt in der Regel durch Blockschaltbilder. Die Beschreibung eines Modellblockes erfolgt anhand der Gegebenheiten des zu modellierenden Teilsystems, wobei das Übertragungsverhalten des Teilsystems durch eine Gleichung bzw. ein Gleichungssystem abgebildet wird [138]. Bei Werkzeugmaschinen sind die Wirkstrukturen der Regelungen inklusive der Stelleinrichtungen durch die verbreiteten antriebstechnischen Lösungen mit kaskadierter Lageregelung weitgehend festgelegt [97, 98]. Für die Modellbildung dieser Teilsysteme bedeutet dies, dass die Modellblöcke einmalig konfiguriert und zur Simulation einer Maschine nur noch parametrisiert werden müssen [138].

Die Modellbildung des mechanischen Teils erfolgt durch Strukturmodelle, wobei man bei der Betrachtung von Werkzeugmaschinen von Systemen gelenkig verbundener fester Körper ausgeht. In der Festkörpermechanik kommen zur Modellierung der mechanischen Struktur deshalb verschiedene Verfahren zum Einsatz, die Bewegungsgleichungen in Form von gewöhnlichen Differenzialgleichungen liefern. Im wesentlichen kommen dabei drei Verfahrensarten zum Einsatz.

Bei den Verfahren handelt es sich um Starrkörperansätze, um die Finite-Elemente-Methode und um Kombinationen dieser zwei Verfahren den sogenannten elastischen Mehrkörpersystemen. Die Verfahren unterscheiden sich durch die Dimension und der Art der erzeugten Gleichungssysteme. Grundsätzlich lassen sich die Verfahren in solche unterteilen, die nur kleine lineare Bewegungen und solche, die auch große nichtlineare Bewegungen berücksichtigen. Die numerische Qualität der erzeugten Simulationsmodelle werden durch den gewählten Ansatz des Verfahrens bestimmt, wobei einer durch hohe Systemordnung und der Berücksichtigung von Nichtlinearitäten erreichten hohen Abbildungsgenauigkeit entsprechend lange Berechnungszeiten gegenüberstehen [8].

Bei den Starrkörperansätzen erfolgt die Modellierung einer mechanischen Struktur als Mehrkörpersystem (MKS), wobei das Mehrkörpersystem durch starre Körper mit tensorischen Trägheitseigenschaften idealisiert wird [184]. Die Modellierung einer Maschine mit einem Verfahren mit Starrkörperansatz ist nur für solche Maschinen eine geeignete Beschreibungsform, deren Baugruppen, zumindest in dem betrachteten Frequenzbereich, als ideal starr angenommen werden können.

Die FEM stellt heute im gesamten Maschinenbau ein Standardverfahren für strukturmechanische Berechnungen dar. Sie wird bei Werkzeugmaschinen erfolgreich zur Auslegung und Optimierung der Gestellstruktur eingesetzt [138]. Das Vorgehen der FEM besteht in der Unterteilung einer kontinuierlichen Struktur in eine hohe Anzahl von finiten Elementen (FE) mit regelmäßiger Geometrie. Eine kontinuierliche Struktur wird mit dieser Methode in eine raumdiskrete Darstellung

überführt. Der Diskretisierungsgrad der Struktur wird dabei durch die Anzahl der verwendeten finiten Elemente bestimmt. Während ein hoher Diskretisierungsgrad mit einer Erhöhung der Qualität der Näherungslösung einhergeht, wird allerdings zugleich auch die Anzahl der Freiheitsgrade und damit die Größe des entstehenden Gleichungssystems dadurch erhöht.

Das elastische MKS stellt die umfassendste Beschreibungsform für das dynamische Verhalten von Werkzeugmaschinen dar. Allerdings ist die Erzeugung der Simulationsmodelle im Vergleich mit der FEM erheblich aufwendiger, da zusätzlich zur Modellierung der FEM-Substrukturen der Maschine weitere Arbeitsschritte zur vollständigen Erzeugung eines MKS-Modells vonnöten sind. Die meisten elastischen MKS basieren auf der Synthese von geometrisch nichtlinearen Referenzbewegungen körperfester Bezugssysteme und linear-elastischen Verformungen der Körper [145]. Dies führt zu MKS und FEM kombinierten Modellen. Die elastischen Körper werden dabei als FEM-Substrukturen modelliert und entweder direkt oder über einen Rayleigh-Ritz-Ansatz in das MKS eingebunden [161].

Bei regelungstechnischen Analysen werden im Allgemeinen nur lineare Bewegungen um einen Betriebspunkt betrachtet, da bei einer funktionierenden und im linearen Bereich arbeitenden Regelung die Regeldifferenzen klein sind [138]. Für eine Regelsimulation genügt daher die Berechnung der im Vergleich zu den Führungsbewegungen kleinen Regeldifferenzen mit linear-elastischen Modellen. Aus diesem Grund ist die Verwendung der FEM zur Simulation des Strukturverhaltens in Verbindung mit einer Regelung bereits in verschiedenen Arbeiten gegenüber anderen Ansätzen bevorzugt worden, siehe dazu [26, 27, 56, 187] und wird auch in Hinblick auf die industrielle Anwendung als geeigneteres Verfahren erachtet [83].

Darauf aufbauend wird in [138] zur Untersuchung der Einflüsse der Reglerdynamik auf die strukturmechanischen Komponenten und zur Optimierung des mechatronischen Gesamtsystems ein integriertes Verfahren zur Simulation des dynamischen Verhaltens von Werkzeugmaschinen mit elektromechanischen Antriebssystemen entwickelt. Das Verfahren dient der Analyse von Maschinenentwürfen nach strukturmechanischen und regelungstechnischen Kriterien. In einem ersten Schritt erfolgt dabei die FEM-Modellierung des mechanischen Systems zur Beschreibung der statischen und dynamischen Übertragungseigenschaften sowie der elastischen Verformungen. Die mechanischen Komponenten der Antriebe, d.h. die Rotoren und Statoren im Motorgehäuse werden dabei detailliert im FEM-Modell abgebildet. Wechselwirkungen der Gestell- und Antriebsstrukturen werden dabei durch ein integriertes strukturelastisches Modell repräsentiert. In einem zweiten Schritt erfolgt die Modellbildung in blockorientierter Darstellungsform. Das elektro-dynamische Übertragungsverhalten der Servomotoren kann durch einfache Ersatzmodelle mit PT1-Verhalten zusammen mit dem Modell des Transistorstellers im regelungstechnischen Teil des Gesamtmodells berücksichtigt werden. Das nach der Kopplung resultierende Gleichungssystem beschreibt das Verhalten der Regelung unter Berücksichtigung der strukturdynamischen Eigenschaften sowohl der Antriebsstränge als auch des Maschinengestells. Dieses kann zur Simulation des mechatronischen Gesamtsystems und damit zur Analyse der mit den Antrieben realisierbaren Reglerbandbreiten und der Ursachen von Bandbreitenbegrenzungen verwendet werden [138].

1.1.3 Modellreduktion

Die klassischen Verfahren zur Modellreduktion wurden bereits 1965 vorgestellt. Diese basieren auf einer Kondensation der Steifigkeitsmatrix oder der Überlagerung von Eigenmoden. Die Modellreduktion nach Guyan [88] erweitert die statische Kondensation auf dynamische Systeme, wobei das statische Verhalten der Struktur in einer reduzierten Steifigkeitmatrix exakt abgebildet wird. Bei diesem Verfahren werden allerdings in der Massenmatrix gemischte Massen- und Steifigkeitsterme erzeugt, wodurch die Eigenwerte und Eigenvektoren des dynamischen Systems mit großen nicht quantifizierbaren Fehlern belegt werden.

Um eine geeignete Approximation des dynamischen Verhaltens durch ein reduziertes Modell zu erzielen, bieten sich die modalen Verfahren an. Diese Verfahren führen eine Reduktion von Freiheitsgraden im modalen Koordinatenraum nach der Lösung des Eigenwertproblems durch. Dabei werden die orthonormierten Eigenvektoren der Originalstruktur berechnet und als räumliche Deformationskoordinaten verwendet. Die Strukturverformungen der Gesamtstruktur entstehen somit durch Überlagerungen von Eigenvektoren, dieses Vorgehen wird modale Synthese genannt. In Abschnitt 8.3 wird dieses Verfahren detaillierter dargestellt. Da für die verallgemeinerte Koordinatenbasis nur eine beschränkte Anzahl der dynamischen Moden verwendet wird, kann das statische Verhalten des reduzierten Systems mit einem signifikanten Fehler behaftet sein, der den statischen Residuen der nicht berücksichtigten Moden entspricht. Um diesen Fehler zu minimieren, wird in [47] ein Verfahren vorgestellt, das der verallgemeinerten Basis zusätzliche dynamische Korrekturmoden hinzufügt, die durch Rekonstruktion der vernachlässigten Moden berechnet werden. Allerdings ist auch mit Hilfe dieser Methode keine vollständige Kompensation des statischen Fehlers möglich, da hierfür sämtliche Eigenwerte und Eigenvektoren des Originalsystems berechnet werden müssten. Aus dem gleichen Grund ist auch eine Angabe des durch die Modellreduktion erzeugten Fehlers nicht möglich.

Aus der Regelungstechnik und der Kontrolltheorie wurden in den letzten Jahren verschiedene Ansätze zur Modellreduktion bekannt, welche auf der Singulärwertzerlegung und balancierten Abschneiden oder dem Momentenabgleich der Übertragungsfunktion des Modells basieren. Diese Verfahren haben den Vorteil, dass sowohl die statischen als auch die dynamischen Eigenschaften des reduzierten Systems gut approximiert werden können und die Bewertung auf einer mathematischen Grundlage geschieht, wodurch eine Automatisierbarkeit der Reduktion ermöglicht wird. Die Verfahren, die auf der Singulärwertzerlegung und balancierten Abschneiden basieren, beruhen darauf, Zustände zu vernachlässigen, die einerseits viel Energie benötigen, um angeregt zu werden und andererseits nur wenig Ausgangsenergie erzeugen [133]. Die Wahl der reduzierten Dimension des Systems in Zustandsdarstellung kann bei diesen Verfahren anhand der Norm der Hankel-Singulärwerte des reduzierten Systems erfolgen. Ein weiterer Vorteil dieser Verfahren liegt in der Möglichkeit eine a priori berechenbare obere Schranke für den Approximationsfehler des reduzierten Systems angeben zu können. Eine Übersicht der verschiedenen Ausprägungsformen dieser Verfahren findet man u. a. in [10, 86, 65, 188]. Allerdings sind die Verfahren in den Standard-Implementationen für sehr große Systeme numerisch aufwendig, weshalb sie üblicherweise nur für Systeme mit einigen tausend Freiheitsgraden verwendet werden können. Werden Systeme höherer Dimension betrachtet, muß auf Implementationen dieser Verfahren zurückge-

griffen werden, die bezüglich der Reduzierung der Berechnungszeit und der Ausnutzung des zur Verfügung stehenden Speicherplatzes optimiert sind. Zudem wurden diese Verfahren für Systeme erster Ordnung entwickelt, die nicht direkt auf Systeme zweiter Ordnung anwendbar sind. In [132] werden erstmals Verfahren vorgestellt, die die Idee des balancierten Abschneidens verwenden, um Systeme zweiter Ordnung direkt zu reduzieren. Dieser Ansatz wurde in [172] erweitert, wobei für diese Verfahren die Stabilität des reduzierten Systems nicht garantiert und keine Fehlerschranke angegeben werden kann. Eine effiziente Erweiterung dieses Verfahrens ist in [25] zu finden. Eine Methode, die auf der Verwendung von Gramscher Matrizen basiert und sowohl die Erhaltung der Stabilität garantiert als auch die Möglichkeit der Angabe eines Approximationsfehlers erlaubt, ist in [24] zu finden. Dieses Verfahren wurde in [119] um eine frequenzgewichtete Variante erweitert.

Mit den Verfahren, die auf einem Momentenabgleich basieren, können auch sehr hochdimensionale Systeme reduziert werden. Bei diesen Verfahren wird die Potenzreihenentwicklung der Übertragungsfunktion um eine oder mehrere Entwicklungsstellen in der komplexen Zahlenebene betrachtet. Die matrixwertigen Koeffizienten dieser Reihe werden als Momente bezeichnet. Die Approximationsidee dieser Verfahren besteht nun darin, das reduzierte System so zu wählen, dass die ersten Momente des reduzierten Systems mit denen des Originalsystems übereinstimmen, wobei die Approximationsgüte des reduzierten Systems im starken Maße von den gewählten Entwicklungsstellen abhängt. Eine frühe Projektionsmethode zur Realisierung dieser Idee wurde in [49] vorgeschlagen. Numerisch stabilere Verfahren findet man in [67, 81]. Allerdings ist es bei diesen Verfahren im Allgemeinen nicht möglich a priori eine obere Schranke für den Approximationsfehler zu berechnen. Es existieren jedoch Ansätze zur Abschätzung des Approximationsfehlers, die man u.a. in [16, 21, 81] findet.

Da auch diese Verfahren für die Anwendung im Bereich der Kontrolltheorie entwickelt wurden, wird meist von einem System erster Ordnung in Zustandsdarstellung ausgegangen. In [173] wird gezeigt, dass die spezielle Struktur der Krylov-Unterräume, die aus einem System zweiter Ordnung resultieren, ausgenutzt werden kann, um direkt auf Systeme zweiter Ordnung anwendbare Verfahren zu erhalten. Weiterentwicklungen die auf diesem Ansatz beruhen, findet man in [15]. Verfahren, die dabei auf einem sogenannten Krylov-Unterraum zweiter Ordnung zurückgreifen, findet man in [155].

Als neuer Ansatz zur Reduktion von Systemen erster Ordnung auf Basis von Krylov-Unterräumen wurde eine, für die hier betrachtete Problemstellung relevante, Methode vorgestellt, die stabile Single-Input-Single-Output-Systeme (SISO-Systeme) bezüglich der \mathcal{H}_2 -Norm optimal reduziert [87]. Eine Erweiterung dieses Ansatzes stellt ein auf tangentialer Interpolation basiertes Reduktionsverfahren dar [51]. Dieses Verfahren ist mit Krylov-Unterraum-Methoden realisierbar und reduziert Systeme mit mehreren Eingängen und Ausgängen (MIMO-Systeme)² erster Ordnung bezüglich der \mathcal{H}_2 -Norm optimal. Mit dem Verfahren das in [111] vorgestellt wird, ist eine optimale Reduktion unstabiler Systeme erster Ordnung möglich. Optimal reduziert bedeutet im Fall von unstabilen Systemen das Auffinden jenes reduzierten Systems, das den Fehler bezüglich des, um einen Shift-Parameter verschobenen, Originalsystems in der \mathcal{H}_2 -Norm minimiert.

²Englisch: Multiple-Inter-Multiple-Output-Systeme

Im Anwendungsbereich des VLSI³-Design existieren Verfahren zur effizienten Reduktion von MIMO-Systemen erster Ordnung, die auf einer Variante des Arnoldi-Algorithmus basieren [42, 113]. Diese Variante wird globaler Arnoldi-Algorithmus genannt und zeichnet sich bei Anwendung auf Systeme mit vielen Ein- und Ausgängen gegenüber seiner Block-Version durch eine höhere Effizienz aus. Neben einer effektiven Berechnung eines reduzierten Modells erlauben es Verfahren, die auf dieser Variante des Arnoldi-Algorithmus basieren, zudem spezielle Eigenschaften der Reduktionsverfahren für SISO-Systeme auf MIMO-Systeme zu übertragen [43].

1.2 Ziel und Vorgehen der Arbeit

1.2.1 Zielsetzung

Das Ziel dieser Arbeit besteht in der Entwicklung und Validierung automatisierter Verfahren zur Reduktion von strukturmechanischen FEM-Modellen, die für die Simulation von Werkzeugmaschinen ausgelegt werden.

Dabei sollen die mit den entwickelten Verfahren reduzierten Modelle in ihren Eigenschaften so beschaffen sein, dass eine anschließende gekoppelte Simulation mit dem regelungs- und elektrotechnischen Modell der Werkzeugmaschine innerhalb eines Simulationsprogramms möglich ist. Weiterhin werden die folgenden Anforderungen an das Reduktionsverfahren gestellt:

- 1) Der Reduktionsprozess soll automatisch ablaufen. Die Interaktion mit dem Benutzer soll sich auf die Festlegung eines Frequenzintervalls beschränken, in dem die Übertragungsfunktion gut approximiert werden soll.
- 2) Modelle, die durch instabile Systeme beschrieben werden, sollen nach Anwendung der Verfahren in reduzierte Modelle resultieren, die durch stabile Systeme beschrieben werden.
- 3) Nach der erfolgreichen Reduktion eines Systems soll eine Abschätzung des Approximationsfehlers des reduzierten Systems ausgegeben werden.
- 4) Die Reduktion soll auf das strukturmechanische Modell in Form einer Differenzialgleichung zweiter Ordnung anwendbar sein. Dabei soll die Beschreibung des Modells anhand eines Differenzialgleichungssystems zweiter Ordnung erhalten bleiben.

Das Aufstellen der FEM-Modelle erfolgt dabei mit parametrischen FEM-Modellbausteinen, wie sie in [138] vorgestellt werden. Dabei sind die betrachteten Systemmatrizen, die diese Modelle beschreiben, in der Regel hochdimensional und dünn besetzt. Zur Validierung der Reduktionsergebnisse der entwickelten Verfahren wurden diese FEM-Modelle herangezogen.

1.2.2 Vorgehen der Arbeit

Das Vorgehen der vorliegenden Arbeit lässt sich in die zwei Teilschritte untergliedern:

- I) Entwicklung geeigneter Reduktionsverfahren

³Englisch: Very-Large-Scale Integration-Design

- II) Validierung der entwickelten Verfahren im Vergleich mit klassischen Reduktionsverfahren durch Simulation verschiedener mechatronischer Systeme.

Das Vorgehen in Teilschritt I) lässt sich in weitere vier Teilschritte untergliedern:

- 1) Entwicklung von Verfahren zur Reduktion von MIMO-Systemen erster und zweiter Ordnung mit einer geeigneten Methode zur iterativen Entwicklungsstellenwahl basierend auf dem Block-Arnoldi-Algorithmus.
- 2) Entwicklung eines Verfahrens zur effizienten Reduktion von MIMO-Systemen zweiter Ordnung basierend auf dem globalen Arnoldi-Algorithmus.
- 3) Erweiterung der entwickelten Verfahren zur Reduktion von MIMO-Systemen zweiter Ordnung um eine adaptive Bestimmung der Stufe des Krylov-Unterraumes zu jeder Entwicklungsstelle.
- 4) Entwicklung eines automatisierten Verfahrens mit Hilfe einer geeigneten Methode zur Abschätzung des Approximationsfehlers.

Im ersten Schritt erfolgt die Entwicklung einer Methode zur Bestimmung von geeigneten Entwicklungsstellen zur Approximation der Übertragungsfunktionen der in dieser Arbeit betrachteten speziellen Systeme. Dazu wird in dieser Arbeit ein modifiziertes Verfahren, das auf [87] basiert und geeignete Entwicklungsstellen iterativ optimiert, vorgestellt.

Der globale Arnoldi-Algorithmus ist eine Variante des Arnoldi-Algorithmus, der sich im Vergleich mit der Blockversion des Arnoldi-Algorithmus bei der Reduktion von MIMO-Systemen durch eine höhere Effizienz auszeichnet. Im zweiten Schritt werden deshalb Verfahren vorgestellt, die auf dem globalen Arnoldi-Algorithmus basieren, um die Effizienz der in dieser Arbeit entwickelten automatisierten Reduktionsverfahren bei Anwendung auf MIMO-Systeme zu erhöhen.

Für Systeme erster Ordnung wurden in den letzten Jahren Verfahren veröffentlicht, die auf dem globalen Arnoldi-Algorithmus basieren und die Stufe der Krylov-Unterräume zu jeder Entwicklungsstelle adaptiv bestimmen. Aufbauend auf diesen Verfahren werden im dritten Schritt Verfahren zur Reduktion von Systemen zweiter Ordnung entwickelt, die die Stufe des Krylov-Unterraumes an jeder Entwicklungsstelle adaptiv bestimmen.

Nachdem im vierten Schritt eine mögliche Methode zur Abschätzung des Approximationsfehlers des reduzierten Systems dargestellt wird, erfolgt darauf aufbauend die Entwicklung eines automatisierten Verfahrens zur Reduktion von Systemen zweiter Ordnung, das auf dem globalen Arnoldi-Algorithmus basiert.

Das Vorgehen in Teilschritt II) lässt sich in weitere vier Teilschritte untergliedern:

- 1) Modellbildung verschiedener Maschinenstrukturen.
- 2) Modellreduktion des mechanischen Modells, basierend auf MIMO-Systemen zweiter Ordnung, mit den in I) entwickelten Reduktionsverfahren und der modalen Reduktion.
- 3) Modellbildung des regelungs- und elektrotechnischen Teilsystems.

- 4) Analyse der Approximationsergebnisse im Frequenzbereich, Untersuchung von Lastfällen und gekoppelte Regelungssimulation der mechatronischen Gesamtsysteme.

Im ersten Schritt werden Gestell- und Antriebsstrukturen durch ein integriertes strukturelastisches Modell repräsentiert, um die strukturdynamischen Wechselwirkungen dieser Teilsysteme zu simulieren. Für die Modellbildung wird die Finite-Elemente-Methode verwendet, da diese zur strukturmechanischen Berechnung bereits weit verbreitet ist. Nichtlineare Bewegungen werden nicht berücksichtigt, da die zur Systemoptimierung erforderlichen strukturmechanischen Berechnungen und Regelungssimulationen überwiegend im Frequenzbereich durchgeführt oder nur kleine Bewegungen im Zeitbereich betrachtet werden. Für diese Anwendungen sind lineare Formulierungen der Bewegungsgleichungen ausreichend. Zur Erstellung des strukturmechanischen FEM-Gesamtmodells werden die in [138] entwickelten parametrischen Modellbausteine verwendet.

Im zweiten Schritt werden die mechanischen FEM-Modelle, die in der Gestalt von linearen zeitinvarianten Systemen zweiter Ordnung vorliegen zu Systemen erster oder zweiter Ordnung mit kleinerer Dimension reduziert.

Im dritten Schritt erfolgt die Modellbildung des regelungs- und elektrotechnischen Teilsystems. Dafür wurde die für regelungstechnische Systeme übliche blockorientierte Darstellungsform gewählt. Die Wirkstrukturen der Regelung und der elektrischen Antriebe werden dazu als Blockschaltbilder modelliert.

Im vierten Schritt erfolgt eine Validierung der Simulationsergebnisse, die mit den neu entwickelten Verfahren erzielt werden. Dazu werden die Approximationsergebnisse im Frequenzbereich analysiert und spezielle Lastfälle untersucht. Schließlich erfolgt eine gekoppelte Regelungssimulation des reduzierten mechanischen Modells und des regelungs- und elektrotechnischen Modells in der Simulationsumgebung MATLAB/Simulink[®],⁴. Die Regelungssimulation stellt eine Erweiterung der strukturmechanischen Berechnungen dar und verwendet die dazu erzeugten FEM-Modelle zur Beschreibung der Regelstrecke. Dazu werden die mit der FEM erzeugten und in der Dimension reduzierten Systeme in eine regelungstechnische Darstellungsform transformiert und Systemen, die mit den im Maschinenbau üblicherweise verwendeten Verfahren der modalen Reduktion reduziert wurden, gegenüber gestellt.

Da die Modellbildung und die Simulation nicht im Fokus dieser Arbeit liegen, wurde zur Validierung der Reduktionsergebnisse auf FEM-Modelle von Werkzeugmaschinen und deren Modell des regelungs- und elektrotechnischen Teilsystems zurückgegriffen, die am Institut für Werkzeugmaschinen und Betriebswissenschaften (*iwb*) der TU München⁵ modelliert und erstellt wurden.

1.2.3 Aufbau der Arbeit

Im ersten Teil der Arbeit, der die Kapitel zwei bis fünf umfaßt, werden die Grundlagen, die für das Verständnis der Strukturmechanik, der Systemtheorie und der Modellreduktion benötigt werden,

⁴MATLAB/Simulink ist eine kommerzielle, plattformunabhängige Software des Unternehmens The MathWorks, Inc.

⁵Institut für Werkzeugmaschinen und Betriebswissenschaften (*iwb*), Technische Universität München, Boltzmannstr. 15, 85747 Garching

dargestellt. Der zweite Teil, der die Kapitel sechs und sieben umfaßt, stellt den Aufbau und die Funktionsweise sowie die Simulation der in dieser Arbeit betrachteten Werkzeugmaschinen dar. Im dritten Teil, der die Kapitel acht bis elf umfaßt, wird die Entwicklung und Validierung des für die Simulation von Werkzeugmaschinen entwickelte Reduktionsverfahrens beschrieben.

In Kapitel 2 wird eine Übersicht, der zum Verständnis dieser Arbeit benötigten Grundlagen der Linearen Algebra gegeben.

Die Beschreibung der Modelle der regelungs- und elektrotechnischen sowie der mechanischen Teilsysteme erfolgt in Form von Differenzialgleichungssystemen erster oder zweiter Ordnung. In Kapitel 3 erfolgt deshalb eine Darstellung wichtiger systemtheoretischer Grundlagen, die zum Verständnis der Reduktionsverfahren benötigt werden.

In Kapitel 4 werden daraufhin die Grundlagen der Strukturdynamik, wie sie zum Verständnis der in dieser Arbeit verwendeten mechanischen Strukturen benötigt werden, dargestellt.

Die Reduktion von mechanischen Strukturmodellen stellt den zentralen Aspekt der vorliegenden Arbeit dar. In Kapitel 5 erfolgt dazu eine Einführung in die Grundlagen der Modellreduktion von Systemen erster und zweiter Ordnung.

In Kapitel 6 wird die Ausprägung der in dieser Arbeit betrachteten Werkzeugmaschinen dargestellt. Dabei wird auf den Aufbau und die Funktionsweise der elektrischen Antriebe, der mechanischen Übertragungssysteme und der Antriebsregelung von Werkzeugmaschinen näher eingegangen. Zum Abschluß des Kapitels wird das dynamische Verhalten von Werkzeugmaschinen näher betrachtet.

Die Modellreduktion dient der effizienten Simulation von Werkzeugmaschinen. In Kapitel 7 wird daher das Vorgehen bei der Simulation von Werkzeugmaschinen dargestellt. Dazu wird nach einem Überblick die Modellbildung der Regelung und der elektrischen Antriebe sowie die Modellbildung des mechanischen Systems beschrieben. Darauf folgt die Beschreibung der Anforderungen, die ein Modellreduktionsverfahren im Hinblick auf die Reduktion mechanischer Modelle für die Simulation von Werkzeugmaschinen erfüllen muss. Das Kapitel wird schließlich mit der Beschreibung der Kopplung der Gleichungssysteme des regelungs- und des elektrotechnischen Modells und des mechanischen Modells abgeschlossen.

Kapitel 8 gibt eine Übersicht vorhandener Verfahren zur Modellreduktion. An dieser Stelle werden die klassischen Reduktionsverfahren wie die modale Reduktion genauso erwähnt wie neuere Verfahren, die Systeme bezüglich der \mathcal{H}_2 -Norm optimal reduzieren. Neben den Reduktionsverfahren basierend auf Gramschen Matrizen und auf Krylov-Unterräumen wird auch auf Verfahren basierend auf der Proper Orthogonal Decomposition (POD) eingegangen.

Die in dieser Arbeit entwickelten Verfahren basieren auf Moment-Matching. Da für diese Verfahren keine a priori Fehlerschranken angegeben werden können, werden in Kapitel 9 Methoden zur Fehlerschätzung dargestellt. Darüber hinaus werden adaptive Methoden für Krylov-Unterraum-Verfahren beschrieben. Mit Hilfe dieser Fehlerschätzer und der beschriebenen adaptiven Methoden können daraufhin adaptive Verfahren basierend auf Krylov-Unterräumen entwickelt werden.

Die Modellreduktion von strukturmechanischen Modellen stellt den Hauptaspekt dieser Arbeit dar. In Kapitel 10 erfolgt die Entwicklung iterativer Verfahren zur Modellreduktion der speziellen strukturmechanischen Modelle, die bei der Werkzeugmaschinen-simulation verwendet werden. Im

darauf folgenden Kapitel 11 werden die entwickelten Verfahren anhand von zwei Beispielmotellen von Werkzeugmaschinen validiert.

Eine abschließende Zusammenfassung der wesentlichen Aspekte dieser Arbeit wird in Kapitel 12 gegeben.

Teil I

Grundlagen

Kapitel 2

Hilfsmittel der linearen Algebra

Verfahren zur Modellreduktion verwenden vielfach Methoden und Konzepte aus der linearen Algebra. In diesem Kapitel soll deshalb eine Auswahl der für das Verständnis dieser Arbeit relevanten Grundlagen der linearen Algebra dargestellt werden.

In vielen Verfahren der Modellreduktion kommen die Eigen- und die Singulärwertzerlegung zur Anwendung. Deshalb werden diese in Abschnitt 2.1 vorgestellt.

Zur Abschätzung des Approximationsfehlers von reduzierten Modellen ist die Bestimmung der Größe von Elementen von Vektorräumen notwendig. Im zweiten Abschnitt 2.2 erfolgt deshalb eine kurze Zusammenfassung wichtiger Normen für Vektoren und Matrizen. Diese dienen als Grundlage für die in Abschnitt 3.5 betrachteten Normen auf Funktionsräumen.

Im Abschnitt 2.3 erfolgt die Definition unterschiedlicher Matrix-Zerlegungen, die in Algorithmen von Reduktionsverfahren verwendet werden.

In den abschließenden Abschnitten 2.4 und 2.5 erfolgen die Definitionen von Krylov-Unterräumen und von Projektionen, die bei der Modellreduktion eine wichtige Rolle spielen.

Die in diesem Kapitel vorgestellten Begriffe, Definitionen und Konzepte basieren vorwiegend auf [77, 169]. Da es sich hierbei um grundlegende Konzepte der linearen Algebra handelt existieren zahlreiche weitere Veröffentlichungen, die sich zu einer weitergehenden Vertiefung der dargestellten Konzepte anbieten.

2.1 Eigen- und Singulärwertzerlegung

Ein Eigenwertproblem besteht darin, zu einer gegebenen Matrix $\mathbf{A} \in \mathbb{K}^{n \times n}$ Vektoren $\mathbf{x}_i \in \mathbb{C}^n$ und zugehörige Parameter $\lambda_i \in \mathbb{C}$ zu finden, so dass gilt:

$$\mathbf{A} \cdot \mathbf{x}_i = \lambda_i \cdot \mathbf{x}_i \quad \text{für } i = 1, \dots, n. \quad (2.1)$$

Definition 2.1 (Klassisches Eigenwertproblem) $\lambda \in \mathbb{C}$ heißt *Eigenwert* der Matrix \mathbf{A} , falls ein Vektor $\mathbf{0} \neq \mathbf{x} \in \mathbb{C}$ existiert, so dass

$$\mathbf{A} \mathbf{x} = \lambda \mathbf{x} \quad (2.2)$$

gilt.

Der Vektor \mathbf{x} wird Eigenvektor der Matrix \mathbf{A} zum Eigenwert λ und die Suche nach geeigneten Paaren von λ und \mathbf{x} zu einer gegebenen Matrix \mathbf{A} wird klassisches Eigenwertproblem genannt.

Die nichttrivialen Lösungen dieses Eigenwertproblems erhält man durch die Eigenwerte λ_i , $i = 1, \dots, n$, die sich als die Nullstellen des charakteristischen Polynoms $p(\lambda)$ ergeben:

$$p(\lambda) = \det(\mathbf{A} - \lambda \mathbf{I}). \quad (2.3)$$

Korollar 2.1 $\lambda \in \mathbb{C}$ ist genau dann ein Eigenwert von $\mathbf{A} - s\mathbf{I}$, wenn

$$\det(\mathbf{A} - s\mathbf{I}) = 0. \quad (2.4)$$

Einsetzen der Werte λ_i in Gleichung (2.1) liefert zu jedem Eigenwert λ_i den korrespondierenden Eigenvektor \mathbf{x}_i , $i = 1, \dots, n$, der die Gleichung erfüllt. Der i -te Vektor \mathbf{x}_i wird i -ter Eigenvektor genannt.

Existieren n linear unabhängige Vektoren \mathbf{x}_i , $i = 1, 2, \dots, n$ und werden diese spaltenweise zu einer Matrix $\mathbf{X} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \dots \ \mathbf{x}_n] \in \mathbb{K}^{n \times n}$ zusammengefügt, so erhält man eine Ähnlichkeitstransformation

$$\mathbf{X}^{-1} \cdot \mathbf{A} \cdot \mathbf{X} = \mathbf{\Sigma} = \text{diag}(\lambda_1, \dots, \lambda_n), \quad (2.5)$$

wobei die Matrix \mathbf{X} Transformationsmatrix genannt wird und $\mathbf{\Sigma} = \text{diag}(\lambda_1, \dots, \lambda_n)$ eine Diagonalmatrix mit den Eigenwerten auf der Diagonalen bezeichnet. Sind nur $k < n$ der n Vektoren linear unabhängig, kann mit den so genannten Hauptvektoren eine Transformation auf die Jordansche Normalform erreicht werden. Für eine weitergehende Darstellung der Vorgehensweise in diesem Fall sei auf [77] verwiesen.

Nach Multiplikation der Gleichung (2.5) mit \mathbf{X} von links und \mathbf{X}^{-1} von rechts erhält man die sogenannte Eigenwertzerlegung der Matrix \mathbf{A} :

$$\mathbf{A} = \mathbf{X} \cdot \mathbf{\Sigma} \cdot \mathbf{X}^{-1}. \quad (2.6)$$

Das klassische Eigenwertproblem (2.1) lässt sich zum sogenannten verallgemeinerten Eigenwertproblem erweitern. Zu zwei gegebenen Matrizen $\mathbf{A}, \mathbf{B} \in \mathbb{K}^{n \times n}$ werden dabei Vektoren $\mathbf{x}_i \in \mathbb{C}^n$ und zugehörige Parameter $\lambda_i \in \mathbb{C}$ gesucht, so dass gilt:

$$\mathbf{A} \mathbf{x}_i = \lambda_i \mathbf{B} \mathbf{x}_i, \quad \text{für } i = 1, \dots, n. \quad (2.7)$$

Definition 2.2 (Verallgemeinertes Eigenwertproblem) $\lambda \in \mathbb{C}$ heißt endlicher Eigenwert des Matrizenbüschels $\mathbf{A} - s\mathbf{B}$ mit $s \in \mathbb{C}$, falls ein Vektor $\mathbf{x} \neq \mathbf{0}$ existiert, so dass

$$\mathbf{A} \mathbf{x} = \lambda \mathbf{B} \mathbf{x} \quad (2.8)$$

gilt. Man spricht davon, dass das Matrizenbüschel $\mathbf{A} - s\mathbf{B}$ den unendlichen Eigenwert $\lambda = \infty$ besitzt, wenn für einen Vektor $\mathbf{x} \neq \mathbf{0}$ gilt:

$$\mathbf{B} \mathbf{x} = \mathbf{0}. \quad (2.9)$$

Der Vektor wird in beiden Fällen Eigenvektor von $\mathbf{A} - s\mathbf{B}$ zum Eigenwert λ genannt. Die Suche nach geeigneten λ, \mathbf{x} zu gegebenen Matrizen \mathbf{A}, \mathbf{B} wird verallgemeinertes Eigenwertproblem genannt.

Die nicht-trivialen Lösungen dieses Eigenwertproblems erhält man durch die Eigenwerte λ_i , $i = 1, \dots, n$, die sich als die Nullstellen des charakteristischen Polynoms $p(\lambda)$ ergeben:

$$p(\lambda) = \det(\mathbf{A} - \lambda \mathbf{B}). \quad (2.10)$$

Korollar 2.2 $\lambda \in \mathbb{C}$ ist genau dann ein endlicher Eigenwert von $\mathbf{A} - s\mathbf{B}$, wenn

$$\det(\mathbf{A} - s\mathbf{B}) = 0. \quad (2.11)$$

$\lambda = \infty$ ist genau dann Eigenwert von $\mathbf{A} - s\mathbf{B}$, wenn \mathbf{B} singulär ist. Die zugehörigen Eigenvektoren liegen dann im Nullraum von \mathbf{B} .

Sind die Matrizen \mathbf{A} und \mathbf{B} hermitesch (symmetrisch im Reellen) und \mathbf{B} zusätzlich positiv definit so folgt, dass alle Eigenwerte λ_i reell sind und eine Basis aus n linear unabhängigen Eigenvektoren \mathbf{x}_i , $i = 1, \dots, n$ existiert. Sind die Matrizen \mathbf{A} und \mathbf{B} reell, so sind die Eigenvektoren ebenfalls reell. Ist die Matrix \mathbf{A} darüber hinaus ebenfalls positiv definit oder semidefinit, so sind die Eigenwerte positiv bzw. nicht negativ.

Geht man wie im Fall eines Eigenwertproblems (2.1) vor und fügt die n linear unabhängigen Vektoren \mathbf{x}_i , $i = 1, 2, \dots, n$ spaltenweise zu einer Matrix $\mathbf{X} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \dots \ \mathbf{x}_n] \in \mathbb{K}^{n \times n}$ zusammen, so erhält man eine Ähnlichkeitstransformation der Form

$$\mathbf{X}^{-1} \cdot \mathbf{A} \cdot \mathbf{X} = \text{diag}(\lambda_i), \quad (2.12)$$

$$\mathbf{X}^{-1} \cdot \mathbf{B} \cdot \mathbf{X} = \mathbf{I}. \quad (2.13)$$

Eine Erweiterung des Eigenwertproblems stellt das quadratische Eigenwertproblem dar.

Definition 2.3 (Quadratisches Eigenwertproblem) $\lambda \in \mathbb{C}$ heißt endlicher Eigenwert des Matrizenbüschels $\mathbf{A} + s\mathbf{B} + s^2\mathbf{C}$ mit $s \in \mathbb{C}$, falls ein Vektor $\mathbf{x} \neq \mathbf{0}$ existiert, so dass

$$(\mathbf{A} + \lambda \mathbf{B} + \lambda^2 \mathbf{C}) \mathbf{x} = \mathbf{0} \quad (2.14)$$

gilt. Die Suche nach geeigneten λ, \mathbf{x} zu gegebenen Matrizen $\mathbf{A}, \mathbf{B}, \mathbf{C}$ wird quadratisches Eigenwertproblem genannt.

Korollar 2.3 $\lambda \in \mathbb{C}$ ist genau dann ein endlicher Eigenwert von $\mathbf{A} + s\mathbf{B} + s^2\mathbf{C}$, wenn

$$\det(\mathbf{A} + \lambda \mathbf{B} + \lambda^2 \mathbf{C}) = 0 \quad (2.15)$$

gilt.

Anmerkung 2.1 Zur Bestimmung der Eigenwerte werden in der Praxis spezielle numerische Verfahren verwendet, da diese stabiler und effizienter sind als die Bestimmung der Eigenwerte durch Berechnen der Lösungen der Gleichungen (2.4), (2.11) bzw. (2.15).

Ein dem Eigenwertproblem verwandtes Problem ist das Singulärwertproblem. Im Gegensatz zum Eigenwertproblem ist das Singulärwertproblem auch für rechteckige Matrizen definiert. Zu jeder Matrix $\mathbf{S} \in \mathbb{K}^{n \times m}$ existieren unitäre Matrizen $\mathbf{U} \in \mathbb{C}^{n \times n}$ und $\mathbf{V} \in \mathbb{C}^{m \times m}$, so dass

$$\mathbf{S} = \mathbf{U} \cdot \mathbf{\Sigma} \cdot \mathbf{V}^* \quad (2.16)$$

mit der Diagonalmatrix

$$\mathbf{\Sigma} := \left[\begin{array}{ccc|c} \sigma_1 & & & 0 \\ & \ddots & & \\ & & \sigma_k & 0 \\ \hline 0 & & & 0 \end{array} \right] \in \mathbb{R}^{n \times m} \quad (2.17)$$

gilt, siehe [10, 188]. Dabei werden die reellen Einträge der Matrix $\mathbf{\Sigma}$

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_k \geq 0, \quad k = \min\{n, m\} \quad (2.18)$$

als Singulärwerte bezeichnet.

Jedem Singulärwert σ_i sind ein Links- und ein Rechtssingulärvektor zugeordnet. Diese entsprechen dem i -ten Spaltenvektor der Matrix \mathbf{U} bzw. der Matrix \mathbf{V} . Liegt eine reelle Matrix \mathbf{A} vor, so sind die Matrizen \mathbf{U} und \mathbf{V} ebenfalls reell und orthogonal. Aus der Zerlegung folgt, dass

$$\mathbf{S} \cdot \mathbf{v}_i = \sigma_i \mathbf{u}_i, \quad (2.19)$$

$$\mathbf{S}^* \cdot \mathbf{u}_i = \sigma_i \mathbf{v}_i \quad (2.20)$$

gilt. Mit den Gleichungen (2.19) und (2.20) erhält man den folgenden Zusammenhang zwischen der Singulärwertzerlegung und dem Eigenwertproblem (2.1):

$$\underbrace{(\mathbf{S}^* \cdot \mathbf{S})}_{\mathbf{A}} \cdot \underbrace{\mathbf{v}_i}_{\mathbf{x}_i} = \underbrace{\sigma_i^2}_{\lambda_i} \cdot \underbrace{\mathbf{v}_i}_{\mathbf{x}_i}, \quad (2.21)$$

$$\underbrace{(\mathbf{S} \cdot \mathbf{S}^*)}_{\mathbf{A}^*} \cdot \underbrace{\mathbf{u}_i}_{\mathbf{x}_i} = \underbrace{\sigma_i^2}_{\lambda_i} \cdot \underbrace{\mathbf{u}_i}_{\mathbf{x}_i}. \quad (2.22)$$

Die Singulärwertzerlegung bietet viele Interpretationsmöglichkeiten bei der numerischen Analyse von Problemen aus der linearen Algebra. Für die in dieser Arbeit vorgestellten Verfahren zur Modellreduktion ist die folgende Approximationseigenschaft von Bedeutung. Schreibt man eine Matrix \mathbf{A} mit Hilfe der Singulärwertzerlegung als Summe dyadischer Produkte

$$\mathbf{A} = \sum_{i=1}^k \sigma_i \mathbf{u}_i \mathbf{v}_i, \quad (2.23)$$

dann erhält man, durch Abbruch dieser Reihe nach dem s -ten Glied, eine Approximation

$$\tilde{\mathbf{A}} = \sum_{i=1}^s \sigma_i \mathbf{u}_i \mathbf{v}_i \quad (2.24)$$

mit $\text{Rang } s = \text{rang}(\tilde{\mathbf{A}})$.

2.2 Normen

Die Darstellung der Definitionen und Sätze dieses Abschnittes beruht auf Darstellungen, wie sie in [10, 77] zu finden sind. Die Norm eines linearen Raumes \mathcal{V} ist als eine reellwertige Funktion $\|\cdot\|$ auf \mathcal{V} definiert, die besondere Eigenschaften erfüllt.

Definition 2.4 (Norm) Sei \mathcal{V} ein Vektorraum über dem Körper \mathbb{K} der reellen oder komplexen Zahlen. Eine reellwertige Funktion $\|\cdot\| : \mathcal{V} \rightarrow \mathbb{R}_{\geq 0}$ in die nichtnegativen reellen Zahlen heißt Norm auf \mathcal{V} , wenn für alle Vektoren $\mathbf{x}, \mathbf{y} \in \mathcal{V}$ und $\alpha \in \mathbb{K}$ die Eigenschaften

$$\begin{aligned} i) \quad & \|\mathbf{x}\| \geq 0 \quad \text{und} \quad \|\mathbf{x}\| = 0 \text{ genau dann, wenn } \mathbf{x} = 0, \\ ii) \quad & \|\alpha \mathbf{x}\| = |\alpha| \|\mathbf{x}\|, \\ iii) \quad & \|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\| \end{aligned} \quad (2.25)$$

erfüllt sind.

Für Vektoren $\mathbf{x} \in \mathbb{K}^n$ werden häufig die Hölder- oder p -Normen verwendet.

Satz 2.1 (p -Norm) Sei $\mathbf{x} \in \mathbb{K}^n$ und $p \in [1, \infty]$. Die Hölder- oder p -Normen sind definiert als

$$\|\mathbf{x}\|_p = \begin{cases} \left(\sum_{i=1}^n |[x]_i|^p \right)^{\frac{1}{p}}, & \text{für } 1 \leq p < \infty \\ \max_{i \in \{1 \dots n\}} |[x]_i|, & \text{für } p = \infty. \end{cases} \quad (2.26)$$

Für den Fall $p = 2$ erhält man die bekannte Euklidische Norm

$$\|\mathbf{x}\|_2 := \left(\sum_{i=1}^n |[x]_i|^2 \right)^{\frac{1}{2}}. \quad (2.27)$$

Liegen Normen von Matrizen vor, so wird neben den Bedingungen (2.25) zusätzlich die Submultiplikativität:

$$\|\mathbf{A} \cdot \mathbf{B}\| \leq \|\mathbf{A}\| \cdot \|\mathbf{B}\| \quad (2.28)$$

für alle $\mathbf{A}, \mathbf{B}^* \in \mathbb{K}^{n \times m}$ gefordert. Solche Normen sind z.B. die induzierten oder den Vektornormen zugeordnete Matrixnormen, die mit

$$\|\mathbf{A}\| := \sup_{\mathbf{x} \neq 0} \frac{\|\mathbf{A}\mathbf{x}\|}{\|\mathbf{x}\|} \quad (2.29)$$

definiert sind. Aus dieser Definition folgt die Beziehung

$$\|\mathbf{A} \cdot \mathbf{x}\| \leq \|\mathbf{A}\| \cdot \|\mathbf{x}\|. \quad (2.30)$$

Kombinationen von Matrix- und Vektornormen, die diese Bedingung erfüllen, heißen kompatibel oder verträglich.

Die Matrixnormen, die den p -Normen (2.26) zugeordnet werden, werden mit $\|\cdot\|_p$ bezeichnet. Für die Spezialfälle $p = 1, 2$ und ∞ erhält man:

$$\|\mathbf{A}\|_1 := \max_{1 \leq j \leq m} \sum_{i=1}^n |[A]_{ij}| \quad (\text{Spaltensummennorm}), \quad (2.31)$$

$$\|\mathbf{A}\|_2 := \sqrt{\lambda_{\max}(\mathbf{A}^* \mathbf{A})} \quad (\text{Spektralnrm}), \quad (2.32)$$

$$\|\mathbf{A}\|_{\infty} := \max_{1 \leq i \leq n} \sum_{j=1}^m |[A]_{ij}| \quad (\text{Zeilensummennorm}), \quad (2.33)$$

wobei \mathbf{A}^* die Adjungierte der Matrix $\mathbf{A} \in \mathbb{K}^{n \times m}$ darstellt. Neben den induzierten Normen gibt es weitere Matrixnormen. Als Beispiel soll hier die Schatten- p -Norm vorgestellt werden.

Definition 2.5 (Schatten- p -Norm) Seien $\sigma_k(\mathbf{A})$, $k = 1, \dots, \min\{n, m\}$ die Singulärwerte der Matrix $\mathbf{A} \in \mathbb{K}^{n \times m}$. Dann wird

$$\|\mathbf{A}\|_{S,p} = \begin{cases} \left(\sum_{k=1}^{\min\{n,m\}} \sigma_k^p(\mathbf{A}) \right)^{\frac{1}{p}}, & \text{für } 1 \leq p < \infty \\ \max_{1 \leq k \leq \min\{n,m\}} \sigma_k(\mathbf{A}), & \text{für } p = \infty \end{cases} \quad (2.34)$$

die Schatten- p -Norm von \mathbf{A} genannt.

Die Schatten-2-Norm $\|\mathbf{A}\|_{S,2}$ der Matrix \mathbf{A} ist auch als Frobenius- oder Hilbert-Schmidt-Norm bekannt:

$$\|\mathbf{A}\|_F := \|\mathbf{A}\|_{S,2} = \left(\sum_{k=1}^{\min\{n,m\}} \sigma_k^2(\mathbf{A}) \right)^{\frac{1}{2}} = \sqrt{\text{spur}(\mathbf{A}^* \mathbf{A})}. \quad (2.35)$$

Dabei wird

$$\text{spur}(\mathbf{X}) := \sum_{i=1}^n [\mathbf{X}]_{ii} \quad (2.36)$$

die Spur der Matrix $\mathbf{X} \in \mathbb{K}^{m \times m}$ genannt.

Satz 2.2 (Frobenius-Norm) Sei $\mathbf{A} \in \mathbb{K}^{n \times m}$. Die reellwertige Funktion

$$\|\mathbf{A}\|_F := \left(\sum_{i=1}^n \sum_{j=1}^m |[A]_{ij}|^2 \right)^{\frac{1}{2}} = \sqrt{\text{spur}(\mathbf{A}^* \mathbf{A})} \quad (2.37)$$

ist eine Norm auf $\mathbb{K}^{n \times m}$.

Anmerkung 2.2 Die Frobenius-Norm ist keiner Vektornorm zugeordnet, d.h. sie wird durch keine Vektornorm induziert [36]. \square

Mit Hilfe der Singulärwertzerlegung lassen sich die Spektralnorm und die Frobenius-Norm als

$$\|\mathbf{A}\|_2 = \sigma_{\max}, \quad (2.38)$$

$$\|\mathbf{A}\|_F = \left(\sum_{i=1}^{\min(n,m)} \sigma_i^2 \right)^{\frac{1}{2}} \quad (2.39)$$

angeben. Daraus folgt die Beziehung

$$\|\mathbf{A}\|_F \leq \sqrt{\min(n, m)} \|\mathbf{A}\|_2. \quad (2.40)$$

Anmerkung 2.3 Die Approximation (2.24) ist optimal im Sinne der Lösung des Minimierungsproblems

$$\min_{\text{rang}(\tilde{\mathbf{A}})=s} \|\mathbf{A} - \tilde{\mathbf{A}}\|_2 = \sigma_{s+1} \quad (2.41)$$

oder

$$\min_{\text{rang}(\tilde{\mathbf{A}})=s} \|\mathbf{A} - \tilde{\mathbf{A}}\|_F = \left(\sum_{i=s+1}^r \sigma_i^2 \right)^{\frac{1}{2}}, \quad (2.42)$$

siehe z. B. [10]. \square

2.3 Matrix-Zerlegungen

Ein wichtiges Hilfsmittel bei der Modellreduktion ist die Zerlegung einer Matrix in das Produkt von Teilmatrizen, die spezielle Eigenschaften besitzen. In dieser Arbeit werden Verfahren vorgestellt und entwickelt, die eine **LU**-Zerlegung und verschiedene Versionen der **QR**-Zerlegung verwenden. Hier erfolgen ausschließlich die Definitionen der verwendeten Matrix-Zerlegungen, für eine vertiefte Darstellung und für Darstellungen von Algorithmen zur Berechnung der Zerlegungen sei auf [77, 169] verwiesen. Die folgenden Definitionen folgen der Darstellung in [77].

Eine wichtige Matrix-Zerlegung, die bei der Lösung von Gleichungssystemen verwendet wird, ist die **LU**-Zerlegung einer Matrix $\mathbf{A} \in \mathbb{K}^{n \times n}$ in eine untere Dreiecksmatrix \mathbf{L} und eine obere Dreiecksmatrix \mathbf{U} .

Definition 2.6 (LU-Zerlegung) Gegeben sei eine Matrix $\mathbf{A} \in \mathbb{K}^{n \times n}$. Die Zerlegung der Matrix \mathbf{A} der Form

$$\mathbf{A} = \mathbf{L}\mathbf{U} \quad (2.43)$$

in das Produkt einer linken unteren Dreiecksmatrix $\mathbf{L} \in \mathbb{K}^{n \times n}$ und einer rechten oberen Dreiecksmatrix $\mathbf{U} \in \mathbb{K}^{n \times n}$ wird **LU**-Zerlegung der Matrix \mathbf{A} genannt.

Nach der **LU**-Zerlegung der Matrix \mathbf{A} eines Gleichungssystems $\mathbf{A}\mathbf{x} = \mathbf{b}$ mit $\mathbf{b} \in \mathbb{K}^n$ ist, wegen der Struktur der Matrizen \mathbf{L} und \mathbf{U} , die Lösung des Gleichungssystems mit einer Rückwärts- und einer Vorwärtssubstitution effizient zu berechnen.

Anmerkung 2.4 Der Gesamtaufwand zur Berechnung einer Lösung eines Gleichungssystems $\mathbf{A}\mathbf{x} = \mathbf{b}$ setzt sich aus der Berechnung der **LU**-Zerlegung, der Rückwärts- und der Vorwärtssubstitution zusammen. Bei der Berechnung der Lösung eines Gleichungssystems ist dieser Gesamtaufwand mit dem Aufwand einer direkten Berechnung der Lösung zu vergleichen.

Eine weitere wichtige Matrix-Zerlegung ist die **QR**-Zerlegung einer Matrix in eine orthogonale Matrix \mathbf{Q} und eine rechte obere Dreiecksmatrix \mathbf{R} .

Definition 2.7 (QR-Zerlegung) Gegeben sei eine Matrix $\mathbf{A} \in \mathbb{K}^{n \times n}$. Die Zerlegung der Matrix \mathbf{A} der Form

$$\mathbf{A} = \mathbf{Q}\mathbf{R} \quad (2.44)$$

in das Produkt einer orthogonalen Matrix \mathbf{Q} ($\mathbf{Q}^* \mathbf{Q} = \mathbf{I}_n$) und einer rechten oberen Dreiecksmatrix \mathbf{R} wird **QR**-Zerlegung der Matrix \mathbf{A} genannt.

In der vorliegenden Arbeit wird neben der **LU**- und der **QR**-Zerlegung die sogenannte reduzierte oder economy size **QR**-Zerlegung und die Rank-Revealing¹-**QR**-Zerlegung verwendet, die Modifikationen der **QR**-Zerlegung darstellen.

Ist eine Matrix $\mathbf{A} \in \mathbb{K}^{n \times s}$ mit $s < n$ gegeben. Dann bricht die reduzierte **QR**-Zerlegung nach s berechneten Spalten der Matrix \mathbf{Q} und \mathbf{R} ab.

¹Englisch: Rang aufdeckend. Diese Matrix-Zerlegung wird so genannt, da sie den Rang der zerlegten Matrix offen legt.

Definition 2.8 (reduzierte QR -Zerlegung) Sei $A \in \mathbb{K}^{n \times s}$ mit $s < n$ eine Matrix. Die Zerlegung der Matrix A der Form

$$A = QR \quad (2.45)$$

in das Produkt einer orthogonalen Matrix $Q \in \mathbb{K}^{n \times s}$ ($Q^* Q = I_s$) und einer oberen Dreiecksmatrix $R \in \mathbb{K}^{s \times s}$ wird reduzierte QR -Zerlegung der Matrix A genannt.

Bei der Rank-Revealing- QR -Zerlegung erfolgt die Berechnung der QR -Zerlegung mit einer zusätzlichen Spaltenpermutation der Matrix A . Dabei erfolgt die Permutation nach dem Absolutbetrag der Diagonaleinträge der Matrix R . Die Permutation kann dabei mit der vollständigen oder mit der reduzierten QR -Zerlegung kombiniert angewendet werden.

Definition 2.9 (Rank-Revealing- QR -Zerlegung) Sei $A \in \mathbb{K}^{n \times s}$ eine Matrix mit $s \leq n$ und $\text{rang}(A) = \ell$. Die Zerlegung der Matrix A der Form

$$A = QRP^* \quad (2.46)$$

mit

$$R = \begin{bmatrix} R_{11} & R_{12} \\ 0 & 0 \end{bmatrix} \in \mathbb{K}^{s \times s} \quad (2.47)$$

in eine orthogonale Matrix $Q \in \mathbb{K}^{n \times s}$ ($Q^* Q = I_s$), eine Permutationsmatrix $P \in \mathbb{K}^{s \times s}$, eine obere Dreiecksmatrix $R_{11} \in \mathbb{K}^{\ell \times \ell}$ mit $|r_{1,1}| > \dots > |r_{\ell,\ell}|$ und eine Matrix $R_{12} \in \mathbb{K}^{\ell \times (s-\ell)}$ wird Rank-Revealing- QR -Zerlegung der Matrix A genannt.

2.4 Krylov-Unterräume

Eines der wichtigen Konzepte der Modellreduktion basiert auf der Verwendung von geeigneten Krylov-Unterräumen. An dieser Stelle sollen deshalb die Krylov-Unterräume, die in dieser Arbeit verwendet werden, definiert werden.

Definition 2.10 (Krylov-Unterraum) Der Krylov-Unterraum der Stufe k zu einer Matrix $P \in \mathbb{K}^{n \times n}$ und eines (Start-) Vektors $q \in \mathbb{K}^n$ ist definiert als

$$\mathcal{K}_k(P, q) = \text{span}\{q, Pq, P^2q, \dots, P^{k-1}q\}. \quad (2.48)$$

Die linear unabhängigen Vektoren der Sequenz

$$\{q, Pq, P^2q, \dots, P^{k-1}q\} \quad (2.49)$$

werden Basisvektoren des Krylov-Unterraumes genannt.

Wenn der i -te Vektor ($i < k$) der Sequenz (2.49) eine Linearkombination der vorhergehenden Vektoren ist, dann können alle folgenden Vektoren als Linearkombination der ersten $(i - 1)$ Vektoren geschrieben werden. Aus diesem Grund können die ersten i unabhängigen Vektoren als Basis des Krylov-Unterraumes $\mathcal{K}_k(P, q)$ angesehen werden.

Existieren mehr als ein (Start-) Vektor q , verwendet man die Blockversion des oben definierten Krylov-Unterraumes.

Definition 2.11 (Block-Krylov-Unterraum) Der Block-Krylov-Unterraum der Stufe k zu einer Matrix $\mathbf{P} \in \mathbb{K}^{n \times n}$ und einer (Start-) Matrix $\mathbf{Q} \in \mathbb{K}^{n \times m}$ ist definiert als

$$\mathcal{K}_k(\mathbf{P}, \mathbf{Q}) = \text{colspan}\{\mathbf{Q}, \mathbf{P}\mathbf{Q}, \mathbf{P}^2\mathbf{Q}, \dots, \mathbf{P}^{k-1}\mathbf{Q}\}. \quad (2.50)$$

Die linear unabhängigen Vektoren der Sequenz

$$\{\mathbf{Q}, \mathbf{P}\mathbf{Q}, \mathbf{P}^2\mathbf{Q}, \dots, \mathbf{P}^{k-1}\mathbf{Q}\} \quad (2.51)$$

werden Basisvektoren des Block-Krylov-Unterraumes genannt.

Der Block-Krylov-Unterraum $\mathcal{K}_k(\mathbf{P}, \mathbf{Q})$ mit m (Start-) Vektoren kann als Vereinigung vom m Krylov-Unterräumen der Form (2.48) für jeden (Start-) Vektor angesehen werden [77, 153, 169]. Bei der Reduktion von Systemen zweiter Art haben sich die sogenannten Krylov-Unterräume zweiter Ordnung als hilfreich erwiesen, siehe [15, 155].

Definition 2.12 (Block-Krylov-Unterraum zweiter Ordnung) Der Block-Krylov-Unterraum zweiter Ordnung der Stufe k zweier Matrizen $\mathbf{P}_1 \in \mathbb{K}^{n \times n}$, $\mathbf{P}_2 \in \mathbb{K}^{n \times n}$ und einer (Start-) Matrix $\mathbf{Q} \in \mathbb{K}^{n \times m}$ ist definiert als

$$\mathcal{G}_k(\mathbf{P}_1, \mathbf{P}_2, \mathbf{Q}) = \text{colspan}\{\mathbf{X}_0, \mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_{k-1}\}, \quad (2.52)$$

wobei

$$\mathbf{X}_0 = \mathbf{Q}, \quad (2.53)$$

$$\mathbf{X}_1 = \mathbf{P}_1\mathbf{X}_0, \quad (2.54)$$

$$\mathbf{X}_i = \mathbf{P}_1\mathbf{X}_{i-1} + \mathbf{P}_2\mathbf{X}_{i-2}, \quad i = 2, 3, \dots, k-1. \quad (2.55)$$

Die linear unabhängigen Vektoren der Sequenz

$$\{\mathbf{X}_0, \mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_{k-1}\} \quad (2.56)$$

werden Basisvektoren des Block-Krylov-Unterraumes zweiter Ordnung genannt.

2.5 Projektionen

Eine Matrix $\mathbf{P} \in \mathbb{K}^{n \times n}$ wird Projektor auf den Unterraum $\mathcal{V} \subseteq \mathbb{K}^n$ genannt, wenn diese die Gleichungen

$$\mathcal{B}(\mathbf{P}) = \mathcal{V} \quad \text{und} \quad \mathbf{P}^2 = \mathbf{P}. \quad (2.57)$$

erfüllt.

Mit dieser Definition eines Projektors kann jedes Element $\mathbf{x} \in \mathbb{K}^n$ als

$$\mathbf{x} = \mathbf{P}\mathbf{x} + (\mathbf{I} - \mathbf{P})\mathbf{x} \quad (2.58)$$

geschrieben werden. Dies erlaubt eine Zerlegung von \mathbb{K}^n in zwei Unterräume \mathcal{V} und \mathcal{W}

$$\mathbb{K}^n = \text{span}(\mathcal{V} \oplus \mathcal{W}), \quad (2.59)$$

wobei $\mathcal{B}(\mathbf{P}) = \mathcal{V}$ und $\mathcal{N}(\mathbf{P}) = \mathcal{W}$. Der Projektor \mathbf{P} teilt somit den Raum \mathbb{K}^n in die zwei Unterräume \mathcal{V} und \mathcal{W} , siehe [152].

Definition 2.13 [152] Ein orthogonaler Projektor $\mathbf{P} \in \mathbb{K}^{n \times n}$ ist ein Projektor, der auf einen Unterraum \mathcal{V} entlang eines Unterraumes \mathcal{W} abbildet, wobei die beiden Unterräume \mathcal{V} und \mathcal{W} orthogonal zueinander sind. Für diesen Projektor gilt $\mathbf{P}^* = \mathbf{P}$. Für $\mathbf{x} \in \mathbb{K}^n$ gilt dann $\mathbf{P}\mathbf{x} \in \mathcal{V}$ und $(\mathbf{I} - \mathbf{P})\mathbf{x} \in \mathcal{W}$.

Wegen dieser Zerlegung in zwei Unterräume werden zwei Basen benötigt, um eine Matrixdarstellung eines Projektors zu erhalten: eine Basis $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_r]$ für den Unterraum \mathcal{V} und eine Basis $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_r]$ für den Unterraum \mathcal{W} .

Sind diese Basen biorthogonal, d.h. $\mathbf{W}^*\mathbf{V} = \mathbf{I}$, dann gilt für die Matrix des Projektors:

$$\mathbf{P} = \mathbf{V}\mathbf{W}^*. \quad (2.60)$$

Ansonsten gilt für die Matrix des Projektors:

$$\mathbf{P} = \mathbf{V}(\mathbf{W}^*\mathbf{V})^{-1}\mathbf{W}^*. \quad (2.61)$$

Werden zwei unterschiedliche Matrizen \mathbf{V} und \mathbf{W} verwendet, spricht man von einer schiefen Projektion oder *Petrov-Galerkin-Projektion*. Wird hingegen $\mathbf{W} = \mathbf{V}$ mit $\mathbf{V}^*\mathbf{V} = \mathbf{I}$ verwendet, so spricht man von einer orthogonalen Projektion oder *Galerkin Projektion*. In Abbildung 2.1 ist die orthogonale und die schiefe Projektion schematisch dargestellt, siehe [10, 81, 152].

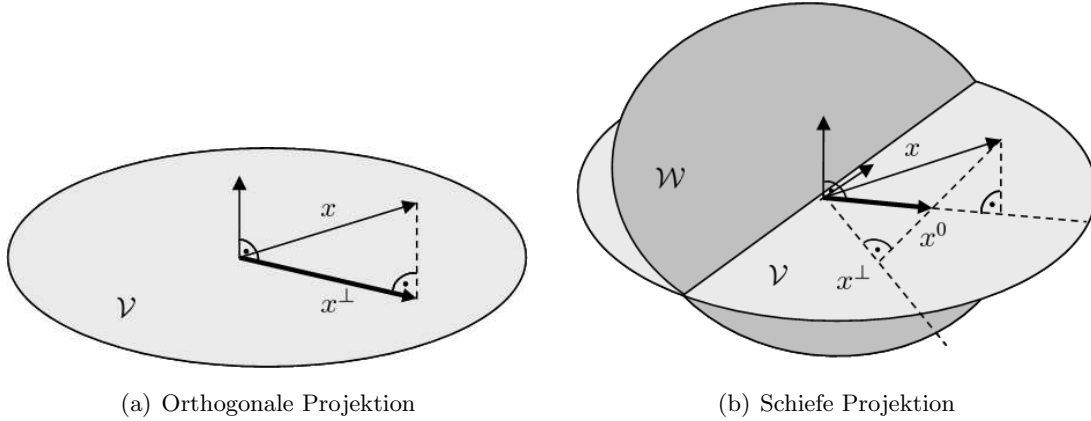


Abbildung 2.1: Schematische Darstellung einer orthogonalen und einer schiefen Projektion, wobei $\mathcal{V} = \text{span}\{\mathbf{v}_1, \dots, \mathbf{v}_r\}$ und $\mathcal{W} = \text{span}\{\mathbf{w}_1, \dots, \mathbf{w}_r\}$

Liegt bei einer Projektion die Matrix $\mathbf{V} = [\mathbf{V}_0 \mathbf{V}_1 \dots \mathbf{V}_{k-1}]$ mit $\mathbf{V}_j \in \mathbb{K}^{n \times s}$ in F -orthonormaler Form vor, d. h. gilt:

$$\langle \mathbf{V}_i, \mathbf{V}_j \rangle_F := \text{spur}(\mathbf{V}_i^* \mathbf{V}_j) = \begin{cases} 0, & \text{für } i, j = 0, 1, \dots, k-1 \text{ mit } i \neq j \\ 1, & \text{für } i = j, \end{cases} \quad (2.62)$$

dann muss anstatt \mathbf{V}^* die Pseudo-Inverse

$$\mathbf{V}^\dagger = (\mathbf{V}^*\mathbf{V})^{-1}\mathbf{V}^* \quad (2.63)$$

verwendet werden, damit $\mathbf{V}^\dagger \mathbf{V} = \mathbf{I}$ gilt. Damit erhält man in diesem Fall für die Matrix des Projektors:

$$\mathbf{P} = \mathbf{V}\mathbf{V}^\dagger = \mathbf{V}(\mathbf{V}^*\mathbf{V})^{-1}\mathbf{V}^*. \quad (2.64)$$

Kapitel 3

Systemtheoretische Grundlagen

In diesem Abschnitt werden wichtige Konzepte der Systemtheorie dargestellt. Diese Kenntnisse sind für das Verständnis, der im folgenden Kapitel beschriebenen regelungstechnischen und mechanischen Systeme als auch der Verfahren zur Modellreduktion, notwendig.

Die meisten in diesem Kapitel zusammengefassten Grundlagen der Systemtheorie orientieren sich an den Darstellungen in einer Vielzahl von Lehrbüchern, wie z.B. [10, 34, 92, 150, 165, 166, 188]. Für eine detailliertere Darstellung dieser Grundlagen sei auf diese Literatur verwiesen.

Nach der Definition linearer zeitinvarianter Systeme und deren Darstellungsformen in Abschnitt 3.1, erfolgt in Abschnitt 3.2 eine Darstellung der Möglichkeiten, das Übertragungsverhalten eines Systems zu beschreiben.

In den folgenden Abschnitten erfolgt daraufhin die Darstellung weiterer wichtiger Eigenschaften von Systemen. Dazu werden in Abschnitt 3.3 die Begriffe der Stabilität und der Passivität von Systemen und in Abschnitt 3.4 das Konzept der Steuerbarkeit und Beobachtbarkeit vorgestellt. Zum Abschluss des Kapitels erfolgt schließlich in Abschnitt 3.5 eine Einführung wichtiger Signal- und Systemnormen.

3.1 Darstellung von Systemen

In diesem Abschnitt wird die Darstellungsform von Strukturmodellen in Form von Systemen definiert. In der Strukturmechanik wird dabei vorwiegend die Darstellungsform eines Systems zweiter Ordnung verwendet, da bei dieser Darstellungsform die physikalischen Eigenschaften der Struktur durch das System repräsentiert werden.

In der Regelungstechnik hat sich hingegen die Darstellungsform als System erster Ordnung (Zustandsraumdarstellung) durchgesetzt.

Man betrachte die drei linearen Räume: den Raum $\mathbb{X} = \{\mathbf{x} : \mathbb{R} \rightarrow \mathbb{K}^n\}$, den Raum $\mathbb{U} = \{\mathbf{u} : \mathbb{R} \rightarrow \mathbb{K}^m\}$ der Eingänge und den Raum $\mathbb{Y} = \{\mathbf{y} : \mathbb{R} \rightarrow \mathbb{K}^q\}$ der Ausgänge, wobei wir uns in dieser Arbeit auf die komplexen ($\mathbb{K} = \mathbb{C}$) und reellen ($\mathbb{K} = \mathbb{R}$) Räume beschränken.

Definition 3.1 (LTI System zweiter Ordnung) Ein lineares zeitinvariantes (LTI)¹ System zweiter Ordnung ist gegeben durch:

$$\Sigma_{2o} := \begin{cases} M\ddot{\mathbf{x}}(t) + D\dot{\mathbf{x}}(t) + K\mathbf{x}(t) = F\mathbf{u}(t), \\ \mathbf{y}(t) = C_p\mathbf{x}(t) + C_v\dot{\mathbf{x}}(t). \end{cases} \quad (3.1)$$

wobei $M, D, K \in \mathbb{K}^{n \times n}$, $F \in \mathbb{K}^{n \times m}$ und $C_p, C_v \in \mathbb{K}^{q \times n}$.

Hierbei ist $t \in \mathbb{R}_+$ die Zeitvariable und man nennt $\mathbf{x}(t) \in \mathbb{K}^n$ die Position oder den inneren Zustand des Systems, $\mathbf{u}(t) \in \mathbb{K}^m$ den Eingang und $\mathbf{y}(t) \in \mathbb{K}^q$ den Ausgang des Systems zum Zeitpunkt t .

Anmerkung 3.1 Die Zeitinvarianz des Systems äußert sich dadurch, dass die Systemmatrizen M, D, K, F, C_p und C_v unabhängig von der Zeitvariablen t sind.

Als Eingangsfunktionen werden stückweise glatte Funktionen $\mathbf{u} : \mathbb{R} \rightarrow \mathbb{K}^m$ mit $\mathbf{u}(t) = \mathbf{0}$, $\forall t < t_0$ betrachtet, die für gewisse $C, \alpha, t_0 \in \mathbb{R}$ der Wachstumsbedingung

$$\|\mathbf{u}(t)\|_2 < Ce^{\alpha t}, \quad \forall t \in [t_0, +\infty] \quad (3.2)$$

unterliegen. Im Fall von beschränkten Funktionen ist $\alpha = 0$.

Im Falle einer regulären Massenmatrix M handelt es sich bei der ersten Gleichung in (3.1) um ein gewöhnliches lineares Differenzialgleichungssystem (DGL) zweiter Ordnung. In diesem Fall führt die Multiplikation der ersten Gleichung in (3.1) von rechts mit M^{-1} auf ein sogenanntes klassisches lineares Differenzialgleichungssystem zweiter Ordnung:

$$\check{\Sigma}_{2o} := \begin{cases} \ddot{\mathbf{x}}(t) = \check{D}\dot{\mathbf{x}}(t) + \check{K}\mathbf{x}(t) + \check{F}\mathbf{u}(t), \\ \mathbf{y}(t) = C_p\mathbf{x}(t) + C_v\dot{\mathbf{x}}(t) \end{cases} \quad (3.3)$$

mit $\check{D} = -M^{-1}D$, $\check{K} = -M^{-1}K$ und $\check{F} = M^{-1}F$. Ist die Massenmatrix M hingegen singulär, handelt es sich bei der ersten Gleichung in (3.1) um ein gewöhnliches lineares Algebra-Differenzialgleichungssystem (DAE)² zweiter Ordnung.

Wird das Modell in der sogenannten Zustandsraumdarstellung benötigt, betrachten wir den Zustandsraum $\mathbb{Z} = \{\mathbf{z} : \mathbb{R} \rightarrow \mathbb{K}^{2n}\}$ mit $(\mathbb{K} = \mathbb{C})$ oder $(\mathbb{K} = \mathbb{R})$. Der Zustandsvektor \mathbf{z} wird aus der Position \mathbf{x} und deren Ableitung, der Geschwindigkeit $\dot{\mathbf{x}}$, definiert als $\mathbf{z} = [\mathbf{x}^T \dot{\mathbf{x}}^T]^T$, gebildet. Mit diesem Zustandsvektor lässt sich das (LTI) System zweiter Ordnung in ein (LTI) System erster Ordnung, der sogenannten verallgemeinerten Zustandsraumdarstellung transformieren:

$$\Sigma_{1o} := \begin{cases} E\dot{\mathbf{z}}(t) = A\mathbf{z}(t) + B\mathbf{u}(t), \\ \mathbf{y}(t) = C\mathbf{z}(t) \end{cases} \quad (3.4)$$

mit

$$\begin{aligned} A &= \begin{bmatrix} \mathbf{0} & N \\ -K & -D \end{bmatrix} \in \mathbb{K}^{2n \times 2n}, & B &= \begin{bmatrix} \mathbf{0} \\ F \end{bmatrix} \in \mathbb{K}^{2n \times m}, \\ C &= [C_p \ C_v] \in \mathbb{K}^{q \times 2n}, & E &= \begin{bmatrix} N & \mathbf{0} \\ \mathbf{0} & M \end{bmatrix} \in \mathbb{K}^{2n \times 2n} \end{aligned} \quad (3.5)$$

¹Englisch: linear time invariant

²Englisch: Differential Algebraic Equation

oder

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} -\mathbf{K} & \mathbf{0} \\ \mathbf{0} & \mathbf{N} \end{bmatrix} \in \mathbb{K}^{2n \times 2n}, & \mathbf{B} &= \begin{bmatrix} \mathbf{F} \\ \mathbf{0} \end{bmatrix} \in \mathbb{K}^{2n \times m}, \\ \mathbf{C} &= [\mathbf{C}_p \ \mathbf{C}_v] \in \mathbb{K}^{q \times 2n}, & \mathbf{E} &= \begin{bmatrix} \mathbf{D} & \mathbf{M} \\ \mathbf{N} & \mathbf{0} \end{bmatrix} \in \mathbb{K}^{2n \times 2n}. \end{aligned} \quad (3.6)$$

Dabei ist \mathbf{N} eine $n \times n$ Matrix, an die nur die Forderung gestellt wird, dass sie nicht singulär ist, die aber oft mit der $n \times n$ -Einheitsmatrix \mathbf{I}_n identifiziert wird, $\mathbf{0}$ ist eine Nullmatrix jeweils geeigneter Dimension und \mathbf{D} im Fall eines ungedämpften Systems gleich Null. Es sind auch andere Linearisierungen möglich, siehe [91, 127, 177]. So führt die Ersetzung der \mathbf{N} -Blöcke in den Gleichungen (3.5) jeweils durch $-\mathbf{K}$ auf symmetrische Systemmatrizen \mathbf{A} und \mathbf{E} , wenn \mathbf{M} und \mathbf{K} symmetrisch sind.

Anmerkung 3.2 Die verallgemeinerte Zustandsraumdarstellung (3.4) wird als regulär bezeichnet, wenn dessen Systemmatrix \mathbf{E} invertierbar ist. Im Falle einer singulären Systemmatrix \mathbf{E} wird auch das System als singulär bezeichnet.

Anmerkung 3.3 Für ein gegebenes lineares System sind die Systemmatrizen $\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}$ bzw. $\mathbf{M}, \mathbf{D}, \mathbf{K}, \mathbf{F}, \mathbf{C}_p, \mathbf{C}_v$ und der Zustandsvektor $\mathbf{z}(t)$ bzw. $\mathbf{x}(t)$ nicht eindeutig bestimmt. D. h. es ist möglich einen neuen Zustandsvektor $\tilde{\mathbf{x}}(t)$ einzuführen, so dass gilt:

$$\mathbf{z}(t) = \mathbf{R} \tilde{\mathbf{z}}(t) \quad \text{bzw.} \quad \mathbf{x}(t) = \mathbf{R} \tilde{\mathbf{x}}(t), \quad (3.7)$$

wobei \mathbf{R} eine nicht-singuläre Transformationsmatrix entsprechender Dimension ist.

Mit diesem neuen Zustandsvektor erhält man ein System mit dem gleichen Eingangs- und Ausgangsverhältnis der Form:

$$\tilde{\Sigma}_{1o} := \begin{cases} \tilde{\mathbf{E}} \dot{\tilde{\mathbf{z}}}(t) = \tilde{\mathbf{A}} \tilde{\mathbf{z}}(t) + \tilde{\mathbf{B}} \mathbf{u}(t), \\ \mathbf{y}(t) = \tilde{\mathbf{C}} \tilde{\mathbf{z}}(t) \end{cases} \quad (3.8)$$

mit

$$\tilde{\mathbf{E}} = \mathbf{R}^{-1} \mathbf{E} \mathbf{R}, \quad \tilde{\mathbf{A}} = \mathbf{R}^{-1} \mathbf{A} \mathbf{R}, \quad \tilde{\mathbf{B}} = \mathbf{R}^{-1} \mathbf{B}, \quad \text{und} \quad \tilde{\mathbf{C}} = \mathbf{C} \mathbf{R} \quad (3.9)$$

bzw.

$$\begin{aligned} \tilde{\mathbf{M}} \ddot{\tilde{\mathbf{x}}}(t) + \tilde{\mathbf{D}} \dot{\tilde{\mathbf{x}}}(t) + \tilde{\mathbf{K}} \tilde{\mathbf{x}}(t) &= \tilde{\mathbf{F}} \mathbf{u}(t), \\ \mathbf{y}(t) &= \tilde{\mathbf{C}}_p \tilde{\mathbf{z}}(t) + \tilde{\mathbf{C}}_v \dot{\tilde{\mathbf{z}}}(t) \end{aligned} \quad (3.10)$$

mit

$$\begin{aligned} \tilde{\mathbf{M}} &= \mathbf{R}^{-1} \mathbf{M} \mathbf{R}, & \tilde{\mathbf{D}} &= \mathbf{R}^{-1} \mathbf{D} \mathbf{R}, & \tilde{\mathbf{K}} &= \mathbf{R}^{-1} \mathbf{K} \mathbf{R}, \\ \tilde{\mathbf{F}} &= \mathbf{R}^{-1} \mathbf{F}, & \tilde{\mathbf{C}}_p &= \mathbf{C}_p \mathbf{R}, & \tilde{\mathbf{C}}_v &= \mathbf{C}_c \mathbf{R}. \end{aligned} \quad (3.11)$$

Dies bedeutet, dass die Systeme algebraisch äquivalent sind, siehe z.B. [10, 155].

Ist die Matrix \mathbf{E} invertierbar, so folgt aus (3.4) die klassische Zustandsraumdarstellung des Systems (3.4):

$$\check{\Sigma}_{1o} := \begin{cases} \dot{\mathbf{z}}(t) = \check{\mathbf{A}}\mathbf{z}(t) + \check{\mathbf{B}}\mathbf{u}(t), \\ \mathbf{y}(t) = \mathbf{C}\mathbf{z}(t) \end{cases} \quad (3.12)$$

mit $\check{\mathbf{A}} = \mathbf{E}^{-1}\mathbf{A}$ und $\check{\mathbf{B}} = \mathbf{E}^{-1}\mathbf{B}$.

In den meisten praktischen Anwendungen ist die Anzahl der Eingänge m und die Anzahl der Ausgänge q bedeutend kleiner als die Dimension des Systems n bzw. $2n$. Gilt $m > 1$ und $q > 1$, dann handelt es sich um ein sogenanntes MIMO-System³. Im Fall $m = q = 1$ handelt es sich um ein SISO-System⁴ und die Systemmatrizen $\mathbf{F}, \mathbf{C}_p, \mathbf{C}_v$ oder \mathbf{B}, \mathbf{C} können als Vektoren $\mathbf{f}, \mathbf{c}_p^T, \mathbf{c}_v^T$ bzw. \mathbf{b}, \mathbf{c}^T geschrieben werden.

Definition 3.2 (Dimension eines Systems) Die Dimension der Systemmatrizen \mathbf{E} und \mathbf{A} eines Systems erster Ordnung bzw. \mathbf{M}, \mathbf{D} und \mathbf{K} eines Systems zweiter Ordnung wird Dimension des Systems genannt.

Die Darstellung eines Systems durch die Gleichungen der Form (3.1) oder der Form (3.4) mit geeigneten Systemmatrizen $\mathbf{M}, \mathbf{D}, \mathbf{K}, \mathbf{F}, \mathbf{C}_p, \mathbf{C}_v$ bzw. $\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}$ und durch den Zustandsvektor $\mathbf{x}(t)$ bzw. $\mathbf{z}(t)$ wird als Realisierung des Systems bezeichnet.

Wegen Anmerkung 3.3 ist die Realisierung eines Systems nicht eindeutig. Aus diesem Grund kann es für ein System mehrere Realisierungen geben, deren Dimension beliebig groß sein kann. Im Gegensatz dazu existiert für die Dimension eines Systems eine untere Grenze, die McMillan-Grad des Systems genannt wird. Eine Realisierung mit minimaler Dimension wird Minimalrealisierung⁵ genannt [10].

Definition 3.3 (Minimalrealisierung) Sei ein System der Form (3.1), (3.3), (3.4) oder (3.12) mit McMillan-Grad n_m gegeben. Dann wird die (nicht eindeutige) Realisierung des Systems der Ordnung n_m Minimalrealisierung genannt.

Aus der Theorie der gewöhnlichen Differenzialgleichungen gilt, dass für ein gegebenen Anfangswert $\mathbf{z}(0) = \mathbf{z}_0$ und einen zulässigen Wert $\mathbf{u} \in \mathbb{U}$ des Einganges, die Lösung $\mathbf{z}(t)$ der ersten Gleichung in (3.12) durch

$$\mathbf{z}(t) = e^{\check{\mathbf{A}}t} \mathbf{z}_0 + \int_0^t e^{\check{\mathbf{A}}(t-s)} \check{\mathbf{B}}\mathbf{u}(s) ds \quad (3.13)$$

für $t \geq 0$ eindeutig bestimmt wird, siehe z.B. [108]. Nach Einsetzen dieser Lösung in die zweite Gleichung von (3.12) erhält man für $t \geq 0$ als Ausgang des Systems den Wert

$$\mathbf{y}(t) = \mathbf{C}e^{\check{\mathbf{A}}t} \mathbf{z}_0 + \mathbf{C} \int_0^t e^{\check{\mathbf{A}}(t-s)} \check{\mathbf{B}}\mathbf{u}(s) ds. \quad (3.14)$$

³Englisch: Multiple-Inter-Multiple-Output-System

⁴Englisch: Single-Inter-Single-Output-System

⁵In Englisch: minimal realization

Im Fall einer verallgemeinerten Zustandsraumdarstellung der Form (3.4) mit singulärer Matrix \mathbf{E} liegt bei der ersten Gleichung von (3.4) ein Algebro-Differenzialgleichungssystem vor. Dieses lässt sich mit Hilfe einer Weierstrass-Kronecker-Transformation in zwei entkoppelte Systeme zerlegen:

$$\dot{\mathbf{z}}_1 = \mathbf{J}\mathbf{z}_1 + \mathbf{B}_1\mathbf{u} \quad (3.15)$$

$$\mathbf{X}\dot{\mathbf{z}}_2 = \mathbf{z}_2 + \mathbf{B}_2\mathbf{u} \quad (3.16)$$

$$\mathbf{y} = \mathbf{C}_1\mathbf{z}_1 + \mathbf{C}_2\mathbf{z}_2. \quad (3.17)$$

Die Matrizen \mathbf{J} und \mathbf{X} besitzen dabei eine aus Jordanblöcken bestehende Diagonalstruktur, siehe [72]. In der Matrix \mathbf{J} sind diese regulär. Das erste System bildet somit ein reguläres System, zu dem das Anfangswertproblem für jeden Anfangswert $\mathbf{z}_1(0) = \mathbf{z}_{0,1}$ gemäß (3.13) lösbar ist.

Die Hauptdiagonale der Matrix \mathbf{X} besteht ausschließlich aus Nullen. Das bedeutet, dass die Matrix \mathbf{X} nilpotent und das zweite System folglich singulär ist. Durch die Anzahl und die Größe der Jordanblöcke in \mathbf{X} wird dann eine gewisse Zahl von Freiheitsgraden für die Wahl des Anfangswertes $\mathbf{z}_2(0) = \mathbf{z}_{0,2}$ festgelegt. Die zugehörigen Lösungen der Systeme bestehen dann aus einer Linearkombination von Dirac-Distributionen und ihren Ableitungen. Demnach muss das Anfangswertproblem für singuläre Systeme nicht mehr für jeden Anfangswert $\mathbf{z}(0) = \mathbf{z}_0 \in \mathbb{K}^{2n}$ eine Lösung besitzen, siehe [78, 109].

3.2 Übertragungsverhalten

Bei vielen Anwendungen ist ausschließlich der Effekt, den die Eingänge auf die Ausgänge bewirken, von Interesse. Aus diesem Grund werden viele physikalische Systeme nicht durch ihren Zustand beschrieben, sondern durch das Übertragungsverhalten der Eingänge auf die Ausgänge des Systems. In diesem Abschnitt werden deshalb wichtige Konzepte zur Analyse des Übertragungsverhaltens von LTI Systemen beschrieben.

3.2.1 Übertragungsfunktion

Ein Ansatz zur Analyse von LTI Systemen ist der Übergang vom Zeitbereich zum Bild- oder Frequenzbereich. Eine Möglichkeit, die Lösung einer Differenzialgleichung zu bestimmen, besteht in der Verwendung der Laplace-Transformation, siehe [52].

Definition 3.4 (Laplace-Transformation) [52] Für eine Funktion $\mathbf{f} : \mathbb{R} \rightarrow \mathbb{K}^n$ und ein $s \in \mathbb{C}$ wird bei Existenz

$$\mathbf{F}(s) := \mathcal{L}[\mathbf{f}(t)](s) = \int_0^{\infty} \mathbf{f}(t)e^{-st} dt \quad (3.18)$$

als Laplace-Transformation von $\mathbf{f}(t)$ bezeichnet. Bei exponentiell abklingenden Funktionen ist $\mathbf{F}(s)$ auch für $s = i\omega$, $\omega \in \mathbb{R}$ definiert und $\mathbf{F}(i\omega)$ entspricht der Fourier-Transformierten von \mathbf{f} .

Satz 3.1 (Inverse Laplace-Transformation) [52] Für eine Funktion $\mathbf{f} : \mathbb{R} \rightarrow \mathbb{K}^n$ und für deren Laplace-Transformation $\mathbf{F}(s) := \mathcal{L}[\mathbf{f}(t)](s)$ wird bei Existenz

$$\mathcal{L}^{-1}\{\mathbf{F}(s)\} = \int_{x-i\infty}^{x+i\infty} \mathbf{F}(s)e^{st} ds \quad (3.19)$$

als inverse Laplace-Transformation von $\mathbf{F}(s)$ bezeichnet. Dabei wird über einen beliebigen Strahl $\{x + i\infty, x - i\infty\}$ parallel zur imaginären Achse mit $x > \alpha$ integriert.

Werden homogene Anfangsbedingungen ($\mathbf{x} = \dot{\mathbf{x}} = \mathbf{0}$) vorausgesetzt, so erhält man aus der Darstellung des Systems zweiter Ordnung im Zeitbereich (3.1) durch das algebraische Gleichungssystem

$$\begin{aligned} s^2 \mathbf{M} \mathbf{X}(s) + s \mathbf{D} \mathbf{X}(s) + \mathbf{K} \mathbf{X}(s) &= \mathbf{F} \mathbf{U}(s), \\ \mathbf{Y}(s) &= \mathbf{C}_p \mathbf{X}(s) + s \mathbf{C}_v \mathbf{X}(s) \end{aligned} \quad (3.20)$$

die Darstellung im Frequenzbereich mit dem komplexwertigen Parameter s und

$$\mathbf{X}(s) = \mathcal{L}[\mathbf{x}(t)](s), \quad (3.21)$$

$$\mathbf{U}(s) = \mathcal{L}[\mathbf{u}(t)](s), \quad (3.22)$$

$$\mathbf{Y}(s) = \mathcal{L}[\mathbf{y}(t)](s) \quad (3.23)$$

als den Laplace-Transformierten von $\mathbf{x}(t)$, $\mathbf{u}(t)$ und $\mathbf{y}(t)$. Die Elimination von $\mathbf{X}(s)$ in (3.20) führt auf

$$\mathbf{Y}(s) = (\mathbf{C}_p + s \mathbf{C}_v) (s^2 \mathbf{M} + s \mathbf{D} + \mathbf{K})^{-1} \mathbf{F} \cdot \mathbf{U}(s) = \mathbf{H}(s) \cdot \mathbf{U}(s). \quad (3.24)$$

Die $(q \times m)$ -wertige Funktion

$$\mathbf{H}(s) = (\mathbf{C}_p + s \mathbf{C}_v) (s^2 \mathbf{M} + s \mathbf{D} + \mathbf{K})^{-1} \mathbf{F} \quad (3.25)$$

wird Übertragungsfunktion des Systems zweiter Ordnung genannt [156]. Anhand dieser Darstellung erkennt man, dass die Übertragungsfunktion $\mathbf{H}(s)$ die Laplace-Transformierten der Eingänge auf die Laplace-Transformierten der Ausgänge abbildet. Man erhält die gleiche Übertragungsfunktion auch mit Hilfe der Darstellung (3.3):

$$\mathbf{H}(s) = (\mathbf{C}_p + s \mathbf{C}_v) (s^2 \mathbf{I} - s \check{\mathbf{D}} - \check{\mathbf{K}})^{-1} \check{\mathbf{F}}, \quad (3.26)$$

sowie mit der allgemeinen Zustandsdarstellung (3.4):

$$\mathbf{H}(s) = \mathbf{C} (s \mathbf{E} - \mathbf{A})^{-1} \mathbf{B}, \quad (3.27)$$

oder mit der Zustandsdarstellung (3.12):

$$\mathbf{H}(s) = \mathbf{C} (s \mathbf{I} - \check{\mathbf{A}})^{-1} \check{\mathbf{B}}, \quad (3.28)$$

siehe [10].

Anmerkung 3.4 Zur Auswertung von $\mathbf{H}(s)$ ist für jede Stelle s je nach Darstellungsform des Systems ein lineares Gleichungssystem der Form $(s^2\mathbf{I} - s\check{\mathbf{D}} - \check{\mathbf{K}}) \cdot \mathbf{x} = \check{\mathbf{F}}$, $(s^2\mathbf{M} - s\mathbf{D} - \mathbf{K}) \cdot \mathbf{x} = \mathbf{F}$, $(s\mathbf{I} - \check{\mathbf{A}}) \cdot \mathbf{x} = \check{\mathbf{B}}$ oder $(s\mathbf{E} - \mathbf{A}) \cdot \mathbf{x} = \mathbf{B}$ mit m Spaltenvektoren auf der rechten Seite zu lösen.

Anmerkung 3.5 Bei der Simulation von Systemen ist oft der sogenannte Frequenzgang von Interesse. Dabei handelt es sich um den Absolutbetrag der Übertragungsfunktion für $s = 2\pi f i$ mit der Frequenz $f \in [f_{\min}, f_{\max}]$. Ferner wird der über f abgetragene Winkel $\angle(\mathbf{H}(2\pi f i))$ Phasengang genannt.

Anmerkung 3.6 Besitzen die bei einer Simulation auftretenden Eingangssignale ein auf das Intervall $[f_{\min}, f_{\max}]$ beschränktes Frequenzspektrum, d.h. ist $\mathbf{U}(2\pi f i) = \mathbf{0}$ für $f \notin [f_{\min}, f_{\max}]$, so folgt aus $\mathbf{Y}(s) = \mathbf{H}(s) \cdot \mathbf{U}(s)$, dass auch das Ausgangssignal ein beschränktes Spektrum besitzt. Daher ist es erstrebenswert, ein System mit kleinerer Dimension zu erzeugen, dessen Frequenz- und Phasengang im gewünschten Frequenzbereich den des Originalsystems gut approximiert, aber numerisch weniger aufwendig zu berechnen ist.

Das folgende Lemma zeigt die Äquivalenz des auf ein System erster Ordnung transformierten Systems mit dem Originalsystem auf.

Lemma 3.1 Die Übertragungsfunktion des durch die Transformation (3.5) oder (3.6) erzeugten Systems erster Ordnung (3.4) ist identisch mit der des Systems zweiter Ordnung (3.1).

BEWEIS Siehe [10, 109, 155]. □

Anmerkung 3.7 Mit den Beziehungen aus (3.5) oder (3.6) ist $(s\mathbf{E} - \mathbf{A})$ genau dann invertierbar, wenn $(s^2\mathbf{M} + s\mathbf{D} + \mathbf{K})$ invertierbar ist, siehe z.B. [155].

Im weiteren werden einige wichtige Eigenschaften der Übertragungsfunktion von LTI-Systemen dargestellt. Die in diesem Kapitel festgestellte Äquivalenz von Systemen erster und zweiter Ordnung gestattet dabei eine Beschränkung der folgenden Definitionen und Aussagen auf Systeme erster Ordnung.

Definition 3.5 (Pole der Übertragungsfunktion eines Systems) [166] Die Eigenwerte des Matrixbüschels $(s\mathbf{E} - \mathbf{A})$ eines in Minimalrealisierung gegebenen Systems der Form (3.4) bzw. der Matrix $\check{\mathbf{A}}$ eines in Minimalrealisierung gegebenen Systems der Form (3.12) werden Pole der Übertragungsfunktion $\mathbf{H}(s)$ des Systems genannt.

Die Übertragungsfunktionen linearer zeitinvarianter MIMO-Systeme sind matrixwertige Funktionen, die für alle $\omega \in \mathbb{C}$ außer in den Eigenwerten definiert sind. Ist die Übertragungsfunktion $\mathbf{H}(s)$ eines Systems in Minimalrealisierung gegeben, so ist jeder Pol von $\mathbf{H}(s)$ ein Eigenwert des Matrixbüschels $(s\mathbf{E} - \mathbf{A})$ bzw. der Matrix $\check{\mathbf{A}}$. In diesem Fall lässt sich die Stabilität des Systems mit den Polen der Übertragungsfunktion in Verbindung bringen. Liegt keine Minimalrealisierung des Systems vor, ist zwar jeder Pol von $\mathbf{H}(s)$ ein Eigenwert des Matrixbüschels $s\mathbf{E} - \mathbf{A}$ bzw. der

Matrix $\check{\mathbf{A}}$, durch entsprechende Wahl der Eingangs- bzw. Ausgangsmatrizen \mathbf{B}, \mathbf{C} ist es jedoch möglich, zu verhindern, dass bestimmte Eigenwerte als Pole von $\mathbf{H}(s)$ in Erscheinung treten.

Ein weiterer wichtiger Begriff sind die Residuen einer Übertragungsfunktion. In [129] werden die Residuen einer Übertragungsfunktion $\mathbf{H}(s)$ wie folgt definiert.

Definition 3.6 (Residuen) Sei $\mathbf{H}(s)$ die Übertragungsfunktion des SISO-Systems der Form (3.12) mit einer nicht hebbaren isolierten Singularität an der Stelle λ . Dann hat $\mathbf{H}(s)$ die eindeutige Laurentreihen-Entwicklung

$$\mathbf{H}(s) = \sum_{k=-\infty}^{\infty} \gamma_k (s - \lambda)^k \quad (3.29)$$

für alle s in einer Umgebung $\mathbb{D}_R(\lambda) := \{s \in \mathbb{C} : 0 < \|s - \lambda\| < R\}$. Der Koeffizient

$$\gamma_{-1} = \text{res}(\mathbf{H}(s), s = \lambda) \quad (3.30)$$

von $\frac{1}{s-\lambda}$ wird Residuum von $\mathbf{H}(s)$ an der Stelle $s = \lambda$ genannt.

Hat $\mathbf{H}(s)$ an der Stelle λ eine hebbare Singularität, dann ist $\text{res}(\mathbf{H}(s), s = \lambda) = \gamma_{-1} = 0$.

Die Bestimmung der Residuen mit Hilfe der Laurentreihen-Entwicklung ist oft sehr aufwendig. Der folgende Satz 3.2 liefert hilfreiche Aussagen zur praktischen Bestimmung der Residuen.

Satz 3.2 (Bestimmung von Residuen)

1. Hat $\mathbf{H}(s)$ an der Stelle λ einen einfachen Pol, dann gilt

$$\text{res}(\mathbf{H}(s), s = \lambda) = \lim_{s \rightarrow \lambda} (s - \lambda) \mathbf{H}(s). \quad (3.31)$$

2. Hat $\mathbf{H}(s)$ an der Stelle λ einen k -fachen Pol, dann gilt

$$\text{res}(\mathbf{H}(s), s = \lambda) = \frac{1}{(k-1)!} \lim_{s \rightarrow \lambda} \frac{d^{k-1}}{ds^{k-1}} ((s - \lambda)^k \mathbf{H}(s)). \quad (3.32)$$

BEWEIS: Siehe [129]. □

Seien λ_k , $k = 1, \dots, 2n$ die paarweise eindeutigen Pole des Systems (3.12) und seien ϕ_k die dazu gehörenden Residuen von $\mathbf{H}(s)$ an den Stellen λ_k , $\phi_k = \text{res}(\mathbf{H}(s), s = \lambda_k)$. Dann erhält man die Darstellung der Übertragungsfunktion $\mathbf{H}(s)$ in der Form:

$$\mathbf{H}(s) = \sum_{k=1}^{2n} \frac{\phi_k}{s - \lambda_k}. \quad (3.33)$$

Betrachtet man ein MIMO-System der Form (3.12) mit ausschließlich einfachen (paarweise eindeutigen) Polen, dann liegt die Matrix $\check{\mathbf{A}}$ in Diagonalgestalt vor, d.h. $\check{\mathbf{A}} = \text{diag}(\lambda_1, \dots, \lambda_{2n})$, wobei $\lambda_i \neq \lambda_j$ für $i \neq j$.

Andernfalls kann die Matrix $\check{\mathbf{A}}$ mit einer Transformation $\mathbf{z} = \mathbf{T}\tilde{\mathbf{x}}$ in die Diagonalgestalt $\tilde{\mathbf{A}}$ gebracht werden, wobei die Spalten der Matrix \mathbf{T} die Eigenvektoren der Matrix $\check{\mathbf{A}}$ sind. Man

erhält damit das transformierte System mit den Systemmatrizen $\tilde{\mathbf{A}} = \mathbf{T}^{-1}\check{\mathbf{A}}\mathbf{T}$, $\tilde{\mathbf{B}} = \mathbf{T}^{-1}\check{\mathbf{B}}$ und $\tilde{\mathbf{C}} = \check{\mathbf{C}}\mathbf{T}$, wobei $\tilde{\mathbf{A}}$ diagonal ist.

Die Eingangs- bzw. Ausgangsmatrizen seien mit

$$\tilde{\mathbf{B}} = [\check{\mathbf{b}}_1 \cdots \check{\mathbf{b}}_m] \quad \text{und} \quad \mathbf{C} = \begin{bmatrix} \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_q \end{bmatrix} \quad (3.34)$$

bezeichnet. Hier sei angemerkt, dass der Spaltenvektor $\mathbf{b}_l = \tilde{\mathbf{B}}\mathbf{e}_l \in \mathbb{K}^{2n}$ den l -ten Eingang und der Zeilenvektor $\mathbf{c}_p = \mathbf{e}_p^T \mathbf{C} \in \mathbb{K}^{2n}$ den p -ten Ausgang des Systems für $l = 1, \dots, m$ bzw. $p = 1, \dots, q$ repräsentiert, wenn \mathbf{e}_j für den j -ten Spalteneinheitsvektor entsprechender Dimension steht. Die Übertragungsfunktion $\mathbf{H}(s)$ ist eine $(q \times m)$ -dimensionale matrixwertige Funktion mit den skalarwertigen Komponenten $H_{pl}(s) = [\mathbf{H}(s)]_{pl}$:

$$\mathbf{H}(s) = \begin{bmatrix} H_{11}(s) & \cdots & H_{1m}(s) \\ \vdots & \ddots & \vdots \\ H_{q1}(s) & \cdots & H_{qm}(s) \end{bmatrix}, \quad \text{wobei} \quad H_{pl}(s) = \mathbf{e}_p^T \mathbf{C} (s\mathbf{I}_{2n} - \tilde{\mathbf{A}})^{-1} \tilde{\mathbf{B}}\mathbf{e}_l \quad (3.35)$$

für $l = 1, \dots, m$ und $p = 1, \dots, q$. Jedes $H_{pl}(s)$ kann dabei als SISO-Übertragungsfunktion mit Eingang $\mathbf{b}_l = \tilde{\mathbf{B}}\mathbf{e}_l$ und Ausgang $\mathbf{c}_p = \mathbf{e}_p^T \mathbf{C}$ angesehen werden.

Die Darstellung der einzelnen Übertragungsfunktionen der Ein- und Ausgänge $H_{pl}(s)$ in der Form (3.33) führt zu

$$H_{pl}(s) = \sum_{k=1}^{2n} \frac{\phi_k^{pl}}{s - \lambda_k}, \quad \text{für } l = 1, \dots, m \text{ und } p = 1, \dots, q. \quad (3.36)$$

Ein Vergleich der Koeffizienten in den Gleichungen (3.35) und (3.36) führt mit

$$\phi_k^{pl} = [\mathbf{c}_p]_k [\check{\mathbf{b}}_l]_k, \quad \text{für } l = 1, \dots, m, \quad p = 1, \dots, q \quad \text{und} \quad k = 1, \dots, 2n \quad (3.37)$$

schließlich zur einfachen Darstellung der Residuen $\text{res}(H_{pl}(s), s = \lambda_k)$ der einzelnen skalarwertigen Übertragungsfunktionen $H_{pl}(s)$.

3.2.2 Impulsantwort

Eine wichtige Rolle bei der Analyse von LTI Systemen spielt neben der Übertragungsfunktion die Impulsantwort eines Systems. Zum Bestimmen der Impulsantwort wird die Dirac-Deltafunktion (Impulsfunktion) verwendet, siehe [181].

Definition 3.7 (Impulsfunktion) Eine Funktion δ mit

$$\delta(x) = \begin{cases} 0, & \text{für } x \neq 0 \\ \text{undefiniert}, & \text{für } x = 0, \end{cases} \quad (3.38)$$

die die Bedingung

$$\int_{-\infty}^{\infty} \delta(x) dx = 1 \quad (3.39)$$

erfüllt, wird Dirac-Deltafunktion oder Impulsfunktion genannt.

Unter der Impulsantwort ist dabei die Antwort des Systems auf alle Eingangsfunktionen $\mathbf{u}(t)$ zu verstehen, so dass für eine gegebene Eingangsfunktion mit Hilfe der Impulsantwort die Ausgangsfunktion bestimmt werden kann.

Definition 3.8 (Impulsantwort) Die $(q \times m)$ -wertige Matrixfunktion

$$\mathbf{h}(t) := \begin{cases} \mathbf{0}, & \text{für } t < 0 \\ \mathbf{C}e^{\mathbf{A}t}\mathbf{B}, & \text{für } t \geq 0 \end{cases} \quad (3.40)$$

wird Impulsantwort des Systems (3.12) mit Anfangswert $\mathbf{x}(0) = \mathbf{0}$ auf den Impuls δ genannt.

Zwischen der Impulsantwort und der Antwort des Systems im Frequenzbereich besteht eine direkte Verbindung. Wird für die Eingangsfunktion $\mathbf{U}(s) = \mathbf{1}$ angenommen, so entspricht $\mathbf{H}(\omega)$, $\omega \in \mathbb{R}$ der Fourier-Transformierten der Ausgangsfunktion $\mathbf{y}(t)$. Die inverse Laplace-Transformierte von $\mathbf{1}$ ist die (komponentenweise verstandene) δ -Distribution. Für die Impulsantwort gilt deshalb:

$$\mathbf{h}(t) = \mathcal{L}^{-1}\{\mathbf{h}(s)\}(t). \quad (3.41)$$

3.2.3 Momente

Eine wichtige Gruppe von Reduktionsverfahren bilden die Verfahren basierend auf dem Momentenabgleich der Übertragungsfunktion an geeigneten Entwicklungsstellen. Deshalb soll an dieser Stelle der Begriff der Momente einer Übertragungsfunktion definiert werden. Die Darstellung der Momente der Übertragungsfunktion von Systemen erster Ordnung orientiert sich dabei an [10] und die Darstellung der Momente der Übertragungsfunktion von Systemen zweiter Ordnung an [174].

3.2.3.1 Momente Systeme erster Ordnung

Die Übertragungsfunktion (3.27) eines Systems der Form (3.4) ist eine $(q \times m)$ -wertige Matrixfunktion, die nach Umformung auch als

$$\mathbf{H}(s) = -\mathbf{C} \cdot (\mathbf{I} - s\mathbf{A}^{-1}\mathbf{E})^{-1} \cdot \mathbf{A}^{-1}\mathbf{B} \quad (3.42)$$

geschrieben werden kann. Unter der Annahme, dass die Matrizen $(\mathbf{I} - s\mathbf{A}^{-1}\mathbf{E})$ und \mathbf{A} des Systems nicht singulär sind und $\|s\mathbf{A}^{-1}\mathbf{E}\| < 1$ für eine geeignete Matrixnorm gilt, kann mit Anwendung der Neumann-Reihenentwicklung

$$(\mathbf{I} - s\mathbf{A}^{-1}\mathbf{E})^{-1} = \sum_{j=0}^{\infty} (\mathbf{A}^{-1}\mathbf{E})^j s^j \quad (3.43)$$

die Übertragungsfunktion $\mathbf{H}(s)$ in eine Potenzreihe

$$\mathbf{H}(s) = \sum_{j=0}^{\infty} -\mathbf{C}(\mathbf{A}^{-1}\mathbf{E})^j \mathbf{A}^{-1}\mathbf{B} s^j = \sum_{j=0}^{\infty} \mathbf{H}_j(0) s^j \quad (3.44)$$

um die Entwicklungsstelle $s_0 = 0$, wobei

$$\mathbf{H}_j(0) = -\mathbf{C}(\mathbf{A}^{-1}\mathbf{E})^j\mathbf{A}^{-1}\mathbf{B} = \frac{1}{j!} \frac{\partial^j \mathbf{H}(s)}{\partial s^j} \Big|_{s=s_0=0} \quad (3.45)$$

als Momente bezeichnet werden, entwickelt werden [81].

Betrachtet man ein System erster Ordnung der Form (3.12) erhält man mit Hilfe der Neumann-Reihenentwicklung analog die Potenzreihe zur Übertragungsfunktion $\mathbf{H}(s) = -\mathbf{C} \cdot (\mathbf{I} - s\check{\mathbf{A}}^{-1})^{-1} \cdot \check{\mathbf{A}}^{-1}\check{\mathbf{B}}$ des Systems um die Entwicklungsstelle $s_0 = 0$.

Definition 3.9 (Momente Systeme erster Ordnung an der Stelle $s_0 = 0$)

1. Sei ein System erster Ordnung der Form (3.4) mit regulärer Matrix \mathbf{A} gegeben. Der j -te Koeffizient

$$\mathbf{H}_j(0) = -\mathbf{C}(\mathbf{A}^{-1}\mathbf{E})^j\mathbf{A}^{-1}\mathbf{B} = \frac{1}{j!} \frac{\partial^j \mathbf{H}(s)}{\partial s^j} \Big|_{s=0}, \quad j = 0, 1, \dots \quad (3.46)$$

der Potenzreihenentwicklung

$$\mathbf{H}(s) = \sum_{j=0}^{\infty} \mathbf{H}_j(0) s^j \quad (3.47)$$

der Übertragungsfunktion

$$\mathbf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} \quad (3.48)$$

um die Entwicklungsstelle $s_0 = 0$ wird das j -te Moment der Übertragungsfunktion des Systems (an der Entwicklungsstelle 0) genannt.

2. Sei ein System erster Ordnung der Form (3.12) mit regulärer Matrix $\check{\mathbf{A}}$ gegeben. Der j -te Koeffizient

$$\mathbf{H}_j(0) = -\mathbf{C}\check{\mathbf{A}}^{-j}\check{\mathbf{A}}^{-1}\check{\mathbf{B}} - \frac{1}{j!} \frac{\partial^j \mathbf{H}(s)}{\partial s^j} \Big|_{s=0}, \quad j = 0, 1, \dots \quad (3.49)$$

der Potenzreihenentwicklung

$$\mathbf{H}(s) = \sum_{j=0}^{\infty} \mathbf{H}_j(0) s^j \quad (3.50)$$

der Übertragungsfunktion

$$\mathbf{H}(s) = \mathbf{C}(s\mathbf{I} - \check{\mathbf{A}})^{-1}\check{\mathbf{B}} \quad (3.51)$$

um die Entwicklungsstelle $s_0 = 0$ wird das j -te Moment der Übertragungsfunktion des Systems (an der Entwicklungsstelle 0) genannt.

Der Vorgang zur Bestimmung der Potenzreihe der Übertragungsfunktion um die Entwicklungsstelle $s_0 = 0$ kann auf beliebige Punkte $s_0 \in \mathbb{C}$ erweitert werden. Dazu sei die Entwicklungsstelle s_0 so gewählt, dass die Matrix $(\mathbf{A} - s_0\mathbf{E})^{-1}$ existiert, dann ergibt sich eine zu (3.27) äquivalente Darstellung als geshiftete⁶ Übertragungsfunktion an der Stelle s :

$$\mathbf{H}(s) = \mathbf{C}((s - s_0)\mathbf{E} - (\mathbf{A} - s_0\mathbf{E}))^{-1}\mathbf{B}. \quad (3.52)$$

⁶Dieser Begriff stammt vom englischen Verb *to shift*, welches sich mit *verschieben* übersetzen lässt. In der numerischen Mathematik wird diese Sprechweise mangels einer treffenderen Übersetzung auch im deutschen Sprachgebrauch verwendet.

Mit Hilfe der geshifteten Übertragungsfunktion (3.52) kann die Entwicklung von $\mathbf{H}(s)$ in der ungeshifteten Form (3.27) um die Entwicklungsstelle $s_0 \in \mathbb{C}$ auch als Entwicklung von $\mathbf{H}(s)$ in der geshifteten Form um die Entwicklungsstelle 0 formuliert werden. Dazu wird die Gleichung (3.52) der geshifteten Übertragungsfunktion zu

$$\mathbf{H}(s) = -\mathbf{C}(\mathbf{I} - (s - s_0)(\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{E})^{-1}(\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{B} \quad (3.53)$$

$$= -\mathbf{C}(\mathbf{I} - \mathbf{T})^{-1}(\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{B} \quad (3.54)$$

umgeformt. Ist die Matrix $(\mathbf{I} - \mathbf{T})$ in Gleichung (3.54) nicht singulär und gilt $\|\mathbf{T}\| < 1$ bezüglich einer geeigneten Matrixnorm, dann erhält man mit Hilfe der Neumann-Reihenentwicklung analog zu (3.44) die Potenzreihenentwicklung um die Entwicklungsstelle s_0 :

$$\begin{aligned} \mathbf{H}(s) &= \sum_{j=0}^{\infty} -\mathbf{C}((\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{E})^j(\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{B} (s - s_0)^j \\ &= \sum_{j=0}^{\infty} \mathbf{H}_j(s_0) (s - s_0)^j, \end{aligned} \quad (3.55)$$

wobei

$$\mathbf{H}_j(s_0) = -\mathbf{C}((\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{E})^j(\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{B} = \frac{1}{j!} \frac{\partial^j \mathbf{H}(s)}{\partial s^j} \Big|_{s=s_0}. \quad (3.56)$$

Definition 3.10 (Momente Systeme erster Ordnung an einer Stelle $s_0 \in \mathbb{C}$)

1. Sei ein System erster Ordnung der Form (3.4) mit regulärer Matrix $(\mathbf{A} - s_0\mathbf{E})$ und $s_0 \in \mathbb{C}$ gegeben. Der j -te Koeffizient

$$\mathbf{H}_j(s_0) = -\mathbf{C}((\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{E})^j(\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{B} = \frac{1}{j!} \frac{\partial^j \mathbf{H}(s)}{\partial s^j} \Big|_{s=s_0}, \quad j = 0, 1, \dots \quad (3.57)$$

der Potenzreihenentwicklung

$$\mathbf{H}(s) = \sum_{j=0}^{\infty} \mathbf{H}_j(s_0) (s - s_0)^j \quad (3.58)$$

der Übertragungsfunktion

$$\mathbf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} \quad (3.59)$$

um die Entwicklungsstelle $s_0 \in \mathbb{C}$ wird das j -te Moment der Übertragungsfunktion des Systems (an der Entwicklungsstelle s_0) genannt.

2. Sei ein System erster Ordnung der Form (3.12) mit regulärer Matrix $(\check{\mathbf{A}} - s_0\mathbf{I})$ und $s_0 \in \mathbb{C}$ gegeben. Der j -te Koeffizient

$$\mathbf{H}_j(s_k) = -\mathbf{C}(\check{\mathbf{A}} - s_0\mathbf{I})^{-j}(\check{\mathbf{A}} - s_0\mathbf{I})^{-1}\check{\mathbf{B}} = \frac{1}{j!} \frac{\partial^j \mathbf{H}(s)}{\partial s^j} \Big|_{s=s_0}, \quad j = 0, 1, \dots \quad (3.60)$$

der Potenzreihenentwicklung

$$\mathbf{H}(s) = \sum_{j=0}^{\infty} \mathbf{H}_j(s_0) (s - s_0)^j \quad (3.61)$$

der Übertragungsfunktion

$$\mathbf{H}(s) = \mathbf{C}(s\mathbf{I} - \check{\mathbf{A}})^{-1}\check{\mathbf{B}} \quad (3.62)$$

um die Entwicklungsstelle $s_0 \in \mathbb{C}$ wird das j -te Moment der Übertragungsfunktion des Systems (an der Entwicklungsstelle s_0) genannt.

3.2.3.2 Momente Systeme zweiter Ordnung

Liegt ein System zweiter Ordnung der Form (3.1) vor, so ist die Bestimmung der Momente der Übertragungsfunktion über das äquivalente Systems erster Ordnung möglich. Alternativ lassen sich die Momente auch direkt über die Übertragungsfunktion der Form (3.25) bestimmen. Durch Umformen der Gleichung (3.25) erhält man

$$\mathbf{H}(s) = (\mathbf{C}_p + s\mathbf{C}_v)(s^2\mathbf{K}^{-1}\mathbf{M} + s\mathbf{K}^{-1}\mathbf{D} + \mathbf{I})^{-1}\mathbf{K}^{-1}\mathbf{F}. \quad (3.63)$$

Nach der Substitution $(s^2\mathbf{K}^{-1}\mathbf{M} + s\mathbf{K}^{-1}\mathbf{D}) = -\mathbf{T}$ in Gleichung (3.63) erhält man

$$\mathbf{H}(s) = (\mathbf{C}_p + s\mathbf{C}_v)(\mathbf{I} - \mathbf{T})^{-1}\mathbf{K}^{-1}\mathbf{F}. \quad (3.64)$$

Unter der Annahme, dass die Matrix $(\mathbf{I} - \mathbf{T})$ nicht singulär ist und $\|\mathbf{T}\| < 1$ für eine geeignete Matrixnorm gilt, kann mit Anwendung der Neumann-Reihenentwicklung

$$(\mathbf{I} - \mathbf{T})^{-1} = \sum_{j=0}^{\infty} \mathbf{T}^j \quad (3.65)$$

um die Entwicklungsstelle $s_0 = 0$ die Übertragungsfunktion $\mathbf{H}(s)$ in eine Potenzreihe

$$\mathbf{H}(s) = \sum_{j=0}^{\infty} \mathbf{H}_j(0)s^j, \quad (3.66)$$

wobei

$$\mathbf{H}_j(0) = \begin{cases} \mathbf{C}_p \boldsymbol{\Theta}_j(0), & \text{für } j = 0 \\ \mathbf{C}_v \boldsymbol{\Theta}_{j-1}(0) + \mathbf{C}_p \boldsymbol{\Theta}_j(0), & \text{für } j = 1, 2, \dots \end{cases} \quad (3.67)$$

mit

$$\boldsymbol{\Theta}_j(0) = \begin{cases} \mathbf{K}^{-1}\mathbf{F}, & \text{für } j = 0 \\ \mathbf{K}^{-1}(-\mathbf{D}\boldsymbol{\Theta}_0(0)), & \text{für } j = 1 \\ \mathbf{K}^{-1}(-\mathbf{D}\boldsymbol{\Theta}_{j-1}(0) - \mathbf{M}\boldsymbol{\Theta}_{j-2}(0)), & \text{für } j = 2, 3, \dots \end{cases} \quad (3.68)$$

entwickelt werden, siehe [174].

Definition 3.11 (Momente Systeme zweiter Ordnung an der Stelle $s_0 = 0$) Sei ein System zweiter Ordnung der Form (3.1) mit regulärer Matrix \mathbf{K} gegeben. Der j -te Koeffizient

$$\mathbf{H}_j(0) = \left. \frac{1}{j!} \frac{\partial^j \mathbf{H}(s)}{\partial s^j} \right|_{s=0}, \quad j = 0, 1, 2, \dots \quad (3.69)$$

mit

$$\mathbf{H}_j(0) = \begin{cases} \mathbf{C}_p \boldsymbol{\Theta}_j(0), & \text{für } j = 0 \\ \mathbf{C}_v \boldsymbol{\Theta}_{j-1}(0) + \mathbf{C}_p \boldsymbol{\Theta}_j(0), & \text{für } j = 1, 2, \dots \end{cases} \quad (3.70)$$

und

$$\boldsymbol{\Theta}_j(0) = \begin{cases} \mathbf{K}^{-1} \mathbf{F}, & \text{für } j = 0 \\ \mathbf{K}^{-1} (-\mathbf{D} \boldsymbol{\Theta}_0(0)), & \text{für } j = 1 \\ \mathbf{K}^{-1} (-\mathbf{D} \boldsymbol{\Theta}_{j-1}(0) - \mathbf{M} \boldsymbol{\Theta}_{j-2}(0)), & \text{für } j = 2, 3, \dots \end{cases} \quad (3.71)$$

der Potenzreihenentwicklung

$$\mathbf{H}(s) = \sum_{j=0}^{\infty} \mathbf{H}_j(0) s^j \quad (3.72)$$

der Übertragungsfunktion

$$\mathbf{H}(s) = (\mathbf{C}_p + s\mathbf{C}_v) (s^2\mathbf{M} + s\mathbf{D} + \mathbf{K})^{-1} \mathbf{F} \quad (3.73)$$

um die Entwicklungsstelle $s_0 = 0$ wird das j -te Moment der Übertragungsfunktion des Systems zweiter Ordnung (an der Entwicklungsstelle 0) genannt.

In Analogie zur Bestimmung der Momente ausgehend von einem System erster Ordnung kann der Vorgang zur Bestimmung der Potenzreihe der Übertragungsfunktion um die Entwicklungsstelle $s_0 = 0$ auf beliebige Entwicklungsstellen $s_0 \in \mathbb{C}$ erweitert werden. Dazu sei die Entwicklungsstelle s_0 so gewählt, dass die Matrix $(s_0^2\mathbf{M} + s_0\mathbf{D} + \mathbf{K})^{-1}$ existiert, dann ergibt sich eine zu (3.25) äquivalente Darstellung als geshiftete Übertragungsfunktion an der Stelle s :

$$\mathbf{H}(s) = (\tilde{\mathbf{C}}_p + (s - s_0)\mathbf{C}_v) ((s - s_0)^2\mathbf{M} + (s - s_0)\tilde{\mathbf{D}} + \tilde{\mathbf{K}})^{-1} \mathbf{F} \quad (3.74)$$

wobei

$$\begin{aligned} \tilde{\mathbf{D}} &= 2s_0\mathbf{M} + \mathbf{D} \\ \tilde{\mathbf{K}} &= s_0^2\mathbf{M} + s_0\mathbf{D} + \mathbf{K} \\ \tilde{\mathbf{C}}_p &= \mathbf{C}_p + s_0\mathbf{C}_v. \end{aligned} \quad (3.75)$$

Mit Hilfe der geshifteten Übertragungsfunktion (3.74) kann die Entwicklung von $\mathbf{H}(s)$ in der ungeshifteten Form (3.25) um die Entwicklungsstelle $s_0 \in \mathbb{C}$ auch als Entwicklung von $\mathbf{H}(s)$ in der geshifteten Form um die Entwicklungsstelle 0 formuliert werden. Dazu wird die Gleichung (3.74) der geshifteten Übertragungsfunktion zu

$$\mathbf{H}(s) = (\tilde{\mathbf{C}}_p + \tilde{s}\mathbf{C}_v) ((s - s_0)^2\tilde{\mathbf{K}}^{-1}\mathbf{M} + (s - s_0)\tilde{\mathbf{K}}^{-1}\tilde{\mathbf{D}} + \mathbf{I})^{-1} \tilde{\mathbf{K}}^{-1}\mathbf{F} \quad (3.76)$$

umgeformt. Nach der Substitution $((s - s_0)^2\tilde{\mathbf{K}}^{-1}\mathbf{M} + (s - s_0)\tilde{\mathbf{K}}^{-1}\tilde{\mathbf{D}}) = -\mathbf{T}$ in Gleichung (3.76) erhält man

$$\mathbf{H}(s) = (\tilde{\mathbf{C}}_p + \tilde{s}\mathbf{C}_v) (\mathbf{I} - \mathbf{T})^{-1} \tilde{\mathbf{K}}^{-1}\mathbf{F}. \quad (3.77)$$

Unter der Annahme, dass die Matrix $(I - T)$ nicht singulär ist und $\|T\| < 1$ für eine geeignete Matrixnorm gilt, kann mit Anwendung der Neumann-Reihenentwicklung

$$(I - T)^{-1} = \sum_{j=0}^{\infty} T^j \quad (3.78)$$

die Übertragungsfunktion $H(s)$ um eine beliebige Entwicklungsstelle $s_0 \in \mathbb{C}$ in eine Potenzreihe

$$H(s) = \sum_{j=0}^{\infty} H_j(s_0)(s - s_0)^j, \quad (3.79)$$

wobei

$$H_j(s_0) = \begin{cases} \tilde{C}_p \Theta_j(s_0), & \text{für } j = 0 \\ C_v \Theta_{j-1}(s_0) + \tilde{C}_p \Theta_j(s_0), & \text{für } j = 1, 2, \dots \end{cases} \quad (3.80)$$

mit

$$\Theta_j(s_0) = \begin{cases} \tilde{K}^{-1} F, & \text{für } j = 0 \\ \tilde{K}^{-1} (-\tilde{D} \Theta_0(s_0)), & \text{für } j = 1 \\ \tilde{K}^{-1} (-\tilde{D} \Theta_{j-1}(s_0) - M \Theta_{j-2}(s_0)), & \text{für } j = 2, 3, \dots \end{cases} \quad (3.81)$$

entwickelt werden.

Definition 3.12 (Momente Systeme zweiter Ordnung an einer Stelle $s_0 \in \mathbb{C}$) Sei ein System zweiter Ordnung der Form (3.1) mit regulärer Matrix K gegeben. Der j -te Koeffizient

$$H_j(s_0) = \frac{1}{j!} \left. \frac{\partial^j H(s)}{\partial s^j} \right|_{s=s_0}, \quad j = 0, 1, \dots \quad (3.82)$$

mit

$$H_j(s_0) = \begin{cases} \tilde{C}_p \Theta_j(s_0), & \text{für } j = 0 \\ C_v \Theta_{j-1}(s_0) + \tilde{C}_p \Theta_j(s_0), & \text{für } j = 1, 2, \dots \end{cases} \quad (3.83)$$

und

$$\Theta_j(s_0) = \begin{cases} \tilde{K}^{-1} F, & \text{für } j = 0 \\ \tilde{K}^{-1} (-\tilde{D} \Theta_0(s_0)), & \text{für } j = 1 \\ \tilde{K}^{-1} (-\tilde{D} \Theta_{j-1}(s_0) - M \Theta_{j-2}(s_0)), & \text{für } j = 2, 3, \dots \end{cases} \quad (3.84)$$

der Potenzreihenentwicklung

$$H(s) = \sum_{j=0}^{\infty} H_j(s_0)(s - s_0)^j \quad (3.85)$$

der Übertragungsfunktion

$$H(s) = (C_p + sC_v)(s^2M + sD + K)^{-1}F \quad (3.86)$$

um die Entwicklungsstelle $s_0 \in \mathbb{C}$ wird das j -te Moment der Übertragungsfunktion des Systems (an der Entwicklungsstelle s_0) genannt.

3.2.3.3 Markov Parameter

Die Momente an der Entwicklungsstelle $s_0 = \infty$ werden Markov Parameter genannt, siehe [10]. Die Potenzreihenentwicklung von $\mathbf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}$ für $s_0 \rightarrow \infty$ ist dabei definiert als

$$\mathbf{H}(s) = \sum_{j=0}^{\infty} \mathbf{H}_j(s_0) s^{-j-1}, \quad (3.87)$$

wobei

$$\mathbf{H}_j(s_0) = \mathbf{C}(\mathbf{E}^{-1}\mathbf{A})^j \mathbf{E}^{-1}\mathbf{B}, \quad j = 0, 1, \dots \quad (3.88)$$

Definition 3.13 (Markov Parameter)

1. Sei ein System erster Ordnung der Form (3.4) mit nichtsingulärer Matrix \mathbf{E} gegeben. Der j -te Koeffizient

$$\mathbf{H}_j = \mathbf{C}(\mathbf{E}^{-1}\mathbf{A})^j \mathbf{E}^{-1}\mathbf{B}, \quad j = 0, 1, \dots \quad (3.89)$$

der Potenzreihenentwicklung

$$\mathbf{H}(s) = \sum_{j=0}^{\infty} \mathbf{H}_j s^{-j-1} \quad (3.90)$$

der Übertragungsfunktion

$$\mathbf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} \quad (3.91)$$

um die Entwicklungsstelle $s = \infty$ wird der j -te Markov-Parameter der Übertragungsfunktion des Systems genannt.

2. Sei ein System erster Ordnung der Form (3.12) mit nicht singulärer Matrix $\check{\mathbf{A}}$ gegeben. Der j -te Koeffizient

$$\mathbf{H}_j = \mathbf{C}(\check{\mathbf{A}}^{-1})^j \check{\mathbf{B}}, \quad j = 0, 1, \dots \quad (3.92)$$

der Potenzreihenentwicklung

$$\mathbf{H}(s) = \sum_{j=0}^{\infty} \mathbf{H}_j s^{-j-1} \quad (3.93)$$

der Übertragungsfunktion

$$\mathbf{H}(s) = \mathbf{C}(s\mathbf{I} - \check{\mathbf{A}})^{-1}\check{\mathbf{B}} \quad (3.94)$$

um die Entwicklungsstelle $s = \infty$ wird der j -te Markov-Parameter der Übertragungsfunktion des Systems genannt.

Der j -te Markov-Parameter kann auch mit der j -ten Ableitung der Impuls-Antwort zur Zeit $t = 0$ identifiziert werden [104]. Der erste Markov-Parameter \mathbf{H}_0 ist somit die Impulsantwort zum Zeitpunkt $t = 0$.

Eine wichtige Rolle in der Systemtheorie spielt die sogenannte Hankel-Matrix.

Definition 3.14 (Hankel-Matrix) [10] Die Matrix \mathcal{H} der Markov-Parameter eines LTI-Systems der Form

$$\mathcal{H} = \begin{bmatrix} \mathbf{H}_0 & \mathbf{H}_1 & \mathbf{H}_2 & \cdots \\ \mathbf{H}_1 & \mathbf{H}_2 & \mathbf{H}_3 & \cdots \\ \mathbf{H}_2 & \mathbf{H}_3 & \mathbf{H}_4 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad (3.95)$$

wird Hankel-Matrix genannt. Diese Matrix hat die Struktur einer symmetrischen Blockmatrix, d.h. $(\mathcal{H})_{ij} = \mathbf{H}_{i+j-2} \in \mathbb{K}^{q \times m}$, für $i, j > 0$ mit einer unendlichen Anzahl an Zeilen und Spalten.

3.3 Stabilität und Passivität

3.3.1 Stabilität

Eine wichtige Eigenschaft eines Systems ist die Stabilität. Betrachtet man das System (3.12), ist dazu das Anfangswertproblem

$$\dot{\mathbf{z}}(t) = \tilde{\mathbf{A}}\mathbf{z}(t), \quad \mathbf{z}(0) = \mathbf{z}_0 \quad (3.96)$$

zu untersuchen. Stabilität bedeutet dann, dass zu jeder beschränkten Anfangsbedingung \mathbf{z}_0 eine für alle $t \geq 0$ beschränkte Lösung $\mathbf{z}(t)$ des Anfangswertproblems (3.96) existiert, siehe dazu [104, 135]. Wenn zusätzlich gilt

$$\lim_{t \rightarrow \infty} \mathbf{z}(t) = 0, \quad (3.97)$$

so spricht man von asymptotischer Stabilität.

Definition 3.15 (Stabilität) [104, 135] Das System (3.4) ist stabil, wenn für $\mathbf{u}(t) \equiv 0$ und jeden zulässigen Anfangswert $\mathbf{z}(0) = \mathbf{z}_0 \in \mathbb{R}^{2n}$ alle Lösungen $\mathbf{z}(t)$ für $t \rightarrow \infty$ beschränkt sind. Wenn alle Lösungen $\mathbf{z}(t)$ dieser Anfangswertprobleme sogar für $t \rightarrow \infty$ verschwinden, nennt man das System asymptotisch stabil.

Eine entscheidende Bedeutung für die Stabilität eines Systems haben die Eigenwerte des Systems.

Satz 3.3 (Stabilität eines Systems) Ein System der Form (3.4) ist

- stabil (grenzstabil) genau dann, wenn kein endlicher Eigenwert des Matrizenbüschels $(s\mathbf{E} - \mathbf{A})$ einen positiven Realteil und zusätzlich alle rein imaginären endlichen Eigenwerte die Vielfachheit 1 besitzen.
- asymptotisch stabil genau dann, wenn alle endlichen Eigenwerte des Matrizenbüschels $(s\mathbf{E} - \mathbf{A})$ negative Realteile besitzen.

BEWEIS: Siehe [79]. □

Für weitergehende Analysen der Stabilität sei auf [10, 164] verwiesen. Kriterien mit deren Hilfe die Stabilität eines Systems ohne direkte Bestimmung der Eigenwerte analysiert werden kann, sind in [135] zu finden.

Liegt ein System zweiter Ordnung vor, lässt sich eine Analyse der Stabilität nach Transformation des Systems in ein System erster Ordnung durchführen. Alternativ lässt sich das Resultat des Satzes 3.3 direkt auf Systeme zweiter Ordnung der Form (3.1) anwenden. Dazu setzt man den Zustandvektor $\mathbf{z} = [\mathbf{x}^T \dot{\mathbf{x}}^T]^T$ und die Matrizen \mathbf{E} und \mathbf{A} aus (3.5) bzw. (3.6) in die Eigenwertgleichung $(\lambda \mathbf{E} - \mathbf{A}) \mathbf{z} = \mathbf{0}$ ein und erhält nach Elimination von $\dot{\mathbf{x}}$ das quadratische Eigenwertproblem der Dimension n :

$$(\lambda^2 \mathbf{M} + \lambda \mathbf{D} + \mathbf{K}) \mathbf{x} = \mathbf{0}. \quad (3.98)$$

Mit der Definition 2.14 des quadratischen Eigenwertproblems und dem Korollar 2.3 kann das Resultat des Satzes 3.3 auf Systeme zweiter Ordnung übertragen werden.

Korollar 3.1 *Das System zweiter Ordnung der Form (3.1) ist genau dann stabil, wenn für alle Nullstellen $\lambda \in \mathbb{C}$ des Polynoms $p(\lambda) := \det(\lambda^2 \mathbf{M} + \lambda \mathbf{D} + \mathbf{K})$ gilt:*

1. $\operatorname{Re}(\lambda) < 0$
2. $\operatorname{Re}(\lambda) = 0 \Rightarrow \lambda$ ist einfach.

Das System zweiter Ordnung der Form (3.1) ist genau dann asymptotisch stabil, wenn die Nullstellen $\lambda \in \mathbb{C}$ von $p(\lambda)$ vollständig in der offenen linken Halbebene von \mathbb{C} liegen, d.h. $\operatorname{Re}(\lambda) < 0$.

Anmerkung 3.8 *Für den Spezialfall $\mathbf{D} = \mathbf{0}$ ergibt sich mit der Substitution $\tilde{\lambda} = \lambda^2$ wieder ein verallgemeinertes Eigenwertproblem. Falls $\mathbf{D} = \alpha \mathbf{M} + \beta \mathbf{K}$ für $\alpha, \beta \in \mathbb{C}$ erhält man das gleiche Resultat mit der Substitution $\tilde{\lambda} = (\lambda^2 + \lambda \alpha) / (\lambda \beta + 1)$.*

In der Mechanik oder der Schwingungslehre bezeichnet man den letzteren Fall als Bequemlichkeitshypothese.

Als weitere Alternative werden in [134, 135] Stabilitätskriterien für Systeme zweiter Ordnung anhand der Struktur der Systemmatrizen abgeleitet, auf die hier nur hingewiesen werden soll.

3.3.2 Passivität

Nach [125] werden Systeme als passiv bezeichnet, wenn sie keine Energie erzeugen, d.h. wenn die dissipierende Energie zu keiner Zeit größer als die erzeugte Energie des Systems ist.

Definition 3.16 (Passivität) [125] *Ein System mit gleicher Anzahl an Ein- und Ausgängen heißt passiv, wenn eine Konstante β existiert, so dass für beliebige $T \geq 0$, Anfangswerte $\mathbf{x}(0) = \mathbf{x}_0$, Eingangsfunktionen $\mathbf{u}(t)$ und zugehörige Ausgangsfunktionen $\mathbf{y}(t)$ gilt:*

$$\int_0^T (\mathbf{u}(t))^T \mathbf{y}(t) dt \geq \beta. \quad (3.99)$$

Korollar 3.2 *Ein System ist passiv, wenn eine nichtnegative Funktion $S : \mathbb{R}^n \rightarrow \mathbb{R}^+$ existiert mit $S(0) = 0$, so dass für beliebige $T \geq 0$, Anfangswerte $\mathbf{x}(0) = \mathbf{x}_0$ und Eingangsfunktionen $\mathbf{u}(t)$*

zugehörige Lösungen $\mathbf{x}(t)$ des Anfangwertproblems und die zugehörigen Ausgangsfunktionen $\mathbf{y}(t)$ gilt:

$$S(\mathbf{x}(T)) \leq S(\mathbf{x}(0)) + \int_0^T (\mathbf{u}(t))^T \mathbf{y}(t) dt. \quad (3.100)$$

BEWEIS: Die Bedingung ist für $\beta := -S(\mathbf{x}(0))$ in trivialer Weise erfüllt. \square

Eine derartige Funktion S wird Speicherfunktion genannt. Mit ihr ist es möglich, zu beschreiben wieviel Energie das System zu einem Zeitpunkt T gespeichert hat. Passivität lässt sich demnach so interpretieren, dass das System keine Energie erzeugen kann, da im Falle eines passiven Systems nach dem obigen Korollar die zum Zeitpunkt T gespeicherte Energie niemals größer als die Summe aus der zum Zeitpunkt $t = 0$ gespeicherten Energie $S(\mathbf{x}(0))$ und im Zeitraum $[0, T]$ zugeführter Energie ist.

Die Passivität eines Systems ist eng verbunden mit dem Konzept der positiven Reellwertigkeit [125].

Definition 3.17 (Positive Reellwertigkeit) Eine matrixwertige Funktion $\mathbf{H} : \mathbb{C} \rightarrow (\mathbb{C} \cup \infty)^{m \times m}$ ist positiv reell, wenn sie folgende Eigenschaften besitzt:

1. $\mathbf{H}(s)$ ist analytisch für alle s in $\{s \in \mathbb{C} \mid \operatorname{Re}(s) > 0\}$.
2. $\mathbf{H}(\bar{s}) = \bar{\mathbf{H}}(s)$ für alle $s \in \mathbb{C}$.
3. $\operatorname{Re}(\mathbf{w}^* \mathbf{H}(s) \mathbf{w}) \geq 0$ für alle $s \in \mathbb{C}$ mit $\operatorname{Re}(s) > 0$ und alle $\mathbf{w} \in \mathbb{C}^m$.

Speziell für lineare Systeme gilt die folgende Äquivalenz.

Satz 3.4 Das System (3.4) ist genau dann passiv, wenn die zugehörige Übertragungsfunktion (3.27) positiv reell ist.

BEWEIS: Siehe [125]. \square

3.4 Steuer- und Beobachtbarkeit

An dieser Stelle sollen die zwei wichtigen Eigenschaften linearer Systeme der Steuerbarkeit und der Beobachtbarkeit dargestellt werden. Die Darstellungen dieses Abschnittes stützen sich dabei auf [10].

3.4.1 Steuerbarkeit

Das Konzept der Steuerbarkeit beschäftigt sich mit der Frage, inwiefern die Zustände $\mathbf{z}(t)$ eines Systems (3.12) durch die Eingänge $\mathbf{u}(t)$ beeinflusst werden können. Nach [188] heißt ein lineares System dann steuerbar, wenn zu jedem Anfangszustand $\mathbf{z}(0) = \mathbf{z}_0$, jedem Zeitpunkt $T > 0$ und jedem Endzustand \mathbf{z}_T eine stückweise stetige Steuerfunktion $\mathbf{u}(t)$ existiert, so dass $\mathbf{z}(T) = \mathbf{z}_T$ gilt.

Definition 3.18 (Steuerbarkeit)

- i) Ein Zustand $\bar{\mathbf{z}} \in \mathbb{Z}$ ist aus der Ruhelage des Systems ansteuerbar, wenn eine Eingangsfunktion $\mathbf{u}(t) \in \mathbb{U}$ endlicher Energie (d.h. $\|\mathbf{u}\|_2^2 < \infty$), und eine endliche Zeit \bar{T} existiert, so dass

$$\bar{\mathbf{z}} = \mathbf{z}(\bar{T}) = \int_0^{\bar{T}} e^{\check{\mathbf{A}}(\bar{T}-s)} \check{\mathbf{B}} \mathbf{u}(s) ds \quad (3.101)$$

gilt.

- ii) Der Unterraum der ansteuerbaren Zustände $\mathbb{Z}^{anst} \subset \mathbb{Z}$ des Systems (3.12) ist die Menge aller ansteuerbaren Zustände des Systems.

- iii) Das System (3.12) ist (vollständig) steuerbar, wenn $\mathbb{Z}^{anst} = \mathbb{Z}$, d.h. jeder Zustand kann mit einer geeigneten Steuerung aus der Ruhelage des Systems erreicht werden.

- iv) Die Matrix

$$\mathcal{R}(\check{\mathbf{A}}, \check{\mathbf{B}}) = [\check{\mathbf{B}} \quad \check{\mathbf{A}}\check{\mathbf{B}} \quad \check{\mathbf{A}}^2\check{\mathbf{B}} \quad \dots \quad \check{\mathbf{A}}^{2n-1}\check{\mathbf{B}} \quad \dots] \quad (3.102)$$

wird unendliche Steuerbarkeitsmatrix des Systems (3.12) genannt.

Man kann zeigen, dass der Rang der Steuerbarkeitsmatrix \mathcal{R} und die Dimension des von ihren Spaltenvektoren aufgespannten Raumes maximal $2n$ ist. D.h. für die praktische Anwendung ist die folgende (endliche) Steuerbarkeitsmatrix

$$\mathcal{R}_{2n}(\check{\mathbf{A}}, \check{\mathbf{B}}) = [\check{\mathbf{B}} \quad \check{\mathbf{A}}\check{\mathbf{B}} \quad \check{\mathbf{A}}^2\check{\mathbf{B}} \quad \dots \quad \check{\mathbf{A}}^{2n-1}\check{\mathbf{B}}] \quad (3.103)$$

von Interesse.

Die Überprüfung der Steuerbarkeit kann für Systeme der Form (3.12) mit Hilfe des Steuerbarkeitskriteriums nach Kalman oder mit dem Hautus-Test vorgenommen werden. Nach dem Kalman-Kriterium ist ein Paar $(\check{\mathbf{A}}, \check{\mathbf{B}})$ steuerbar, wenn

$$\text{rang}\left(\begin{bmatrix} \check{\mathbf{B}} & \check{\mathbf{A}}\check{\mathbf{B}} & \dots & \check{\mathbf{A}}^{\ell-1}\check{\mathbf{B}} \end{bmatrix}\right) = 2n \quad (3.104)$$

gilt, wobei ℓ die Dimension des Zustandsraumes darstellt. Der Hautus-Test besagt, dass ein Paar $(\check{\mathbf{A}}, \check{\mathbf{B}})$ genau dann steuerbar ist, wenn es der Bedingung

$$\text{rang}\left(\begin{bmatrix} \check{\mathbf{A}} - \mu \mathbf{I} & \check{\mathbf{B}} \end{bmatrix}\right) = 2n, \quad \forall \mu \in \mathbb{C} \quad (3.105)$$

genügt.

3.4.2 Beobachtbarkeit

Das Konzept der Beobachtbarkeit untersucht die Kopplung zwischen dem Zustand des Systems $\mathbf{z}(t)$ und seinem Ausgang $\mathbf{y}(t)$. Ein lineares zeitinvariantes System heißt beobachtbar, wenn für jeden Zeitpunkt $T > 0$ der Anfangszustand \mathbf{z}_0 aus dem Zeitverlauf des Ausgangs $\mathbf{y}(t)$ und der Steuerfunktion $\mathbf{u}(t)$ im Intervall $[0, T]$ bestimmt werden kann [188].

Definition 3.19 (Beobachtbarkeit)

- i) Ein Zustand $\bar{z} \neq \mathbf{0}$ wird beobachtbar genannt, wenn \bar{z} aus der Ruhelage des Systems bestimmt werden kann, d.h. für $\mathbf{u}(t) \equiv \mathbf{0}$ existiert ein $t \geq 0$ so, dass

$$\mathbf{y}(t) = \mathbf{C}e^{\check{\mathbf{A}}t}\bar{z} \neq \mathbf{0} \quad (3.106)$$

gilt.

- ii) Der Unterraum der beobachtbaren Zustände $\mathbb{Z}^{beo} \subset \mathbb{Z}$ des Systems (3.12) ist die Menge aller beobachtbaren Zustände des Systems.

- iii) Das System (3.12) ist (vollständig) beobachtbar, wenn $\mathbb{Z}^{beo} = \mathbb{Z}$, d.h. jeder Zustand kann aus dem Zeitverlauf des Ausgangs $\mathbf{y}(t)$ und der Steuerfunktion $\mathbf{u}(t)$ bestimmt werden.

- iv) Die Matrix

$$\mathcal{O}(\mathbf{C}, \check{\mathbf{A}}) = [\mathbf{C}^* \quad \check{\mathbf{A}}^* \mathbf{C}^* \quad (\check{\mathbf{A}}^2)^* \mathbf{C}^* \quad \dots \quad (\check{\mathbf{A}}^{2n-1})^* \mathbf{C}^* \quad \dots]^* \quad (3.107)$$

wird unendliche Beobachtbarkeitsmatrix des Systems (3.12) genannt.

Man kann zeigen, dass der Rang der Beobachtbarkeitsmatrix \mathcal{O} und die Dimension des von ihren Spaltenvektoren aufgespannten Raumes maximal $2n$ ist. D.h. für die praktische Anwendung ist die folgende (endliche) Beobachtbarkeitsmatrix

$$\mathcal{O}_{2n}(\mathbf{C}, \check{\mathbf{A}}) = [\mathbf{C}^* \quad \check{\mathbf{A}}^* \mathbf{C}^* \quad (\check{\mathbf{A}}^2)^* \mathbf{C}^* \quad \dots \quad (\check{\mathbf{A}}^{2n-1})^* \mathbf{C}^*]^* \quad (3.108)$$

von Interesse.

Die Überprüfung der Steuerbarkeit kann auf die Untersuchung der Steuerbarkeit des zu (3.12) dualen oder adjungierten Systems

$$\begin{aligned} \dot{\mathbf{z}}^*(t) &= \check{\mathbf{A}}^* \mathbf{z}^*(t) + \mathbf{C}^* \mathbf{u}^*(t), \\ \mathbf{y}^*(t) &= \check{\mathbf{B}}^* \mathbf{z}^*(t), \end{aligned} \quad (3.109)$$

zurückgeführt werden [10, 44].

Damit ist analog zu (3.104) ein Paar $(\mathbf{C}, \check{\mathbf{A}})$ genau dann beobachtbar, wenn

$$\text{rang} \left(\begin{bmatrix} \mathbf{C} \\ \mathbf{C}\check{\mathbf{A}} \\ \vdots \\ \mathbf{C}\check{\mathbf{A}}^{2n-1} \end{bmatrix} \right) = 2n \quad (3.110)$$

gilt. Das Hautus-Kriterium für Beobachtbarkeit ist dann analog zu (3.105) mit

$$\text{rang} \left(\begin{bmatrix} \check{\mathbf{A}} - \mu \mathbf{I} \\ \mathbf{C} \end{bmatrix} \right) = 2n, \quad \forall \mu \in \mathbb{C} \quad (3.111)$$

gegeben.

Satz 3.5 Sei (3.12) eine Realisierung des Systems (3.4). Diese Realisierung ist eine Minimalrealisierung genau dann, wenn sie steuerbar und beobachtbar ist.

BEWEIS: Siehe [165]. □

3.4.3 Gramsche Matrizen

Eine weitere wichtige Charakterisierungsmöglichkeit der Steuer- und Beobachtbarkeit eines linearen zeitinvarianten Systems der Form (3.12) stellen die Gramschen Matrizen dar. Dazu wird die lineare Abbildung

$$L : \mathcal{L}_2[0, T] \rightarrow \mathbb{R}^{2n}, z(T) = L u(T) = \int_0^T e^{\check{A}(T-t)} \check{B} u(t) dt \quad (3.112)$$

betrachtet. Steuerbarkeit auf dem Intervall $[0, T]$ bedeutet dann, dass es zu jedem Zustand $z \in \mathbb{K}^{2n}$ mindestens eine Steuerfunktion $u(t) \in \mathcal{L}_2[0, T]$ gibt, so dass dieser Zustand, ausgehend von der Ruhelage des Systems, erreicht werden kann. Der Operator L muss also surjektiv sein. Da der Bildraum $\mathcal{B}(L)$ von L endlichdimensional und damit abgeschlossen ist, kommt dies der Forderung gleich, dass das orthogonale Komplement des Bildraumes die leere Menge ist, d.h. $\mathcal{B}(L)^\perp = \emptyset$. Weiterhin gilt nach [126] die Beziehung $\mathcal{N}(L^*) = \mathcal{B}(L)^\perp$, mit $\mathcal{N}(L^*) = \text{kern}(L^*)$ und dem sogenannten adjungierten Operator

$$L^* : \mathbb{K}^{2n} \rightarrow \mathcal{L}_2[0, T], L^* z = \check{B}^* e^{\check{A}^*(T-t)} z. \quad (3.113)$$

Berücksichtigt man weiter, dass $\mathcal{N}(L^*) = \mathcal{N}(LL^*)$ gilt, so kann die Untersuchung der Steuerbarkeit auf eine Ranguntersuchung einer positiv semidefiniten Matrix zurückgeführt werden [166].

Definition 3.20 (Endliche Gramsche Steuerbarkeitsmatrix) Die endliche Gramsche Steuerbarkeitsmatrix $\mathcal{P}(T)$ für den Zeitpunkt $T < \infty$ des linearen Systems (3.12) ist definiert durch

$$\begin{aligned} \mathcal{P}(T) &:= LL^* = \int_0^T e^{\check{A}(T-t)} \check{B} \check{B}^* e^{\check{A}^*(T-t)} dt \\ &= \int_0^T e^{\check{A}t} \check{B} \check{B}^* e^{\check{A}^*t} dt. \end{aligned} \quad (3.114)$$

Mit Verwendung des Kalman-Kriteriums ist ein Matrizen-Paar (\check{A}, \check{B}) somit genau dann auf dem Intervall $[0, T]$ steuerbar, wenn $\mathcal{P}(T) \in \mathbb{K}^{2n \times 2n}$ regulär und damit positiv definit ist.

Mit Hilfe von $\mathcal{P}(T)$ kann auch eine Steuerfunktion $\bar{u}(t)$ angegeben werden, um das betrachtete System aus der Ruhelage des Systems $z(0) = \mathbf{0}$ im Intervall $[0, T]$ in den Zustand z zu überführen:

$$\bar{u}(t) = L^*(LL^*)^{-1} z = \check{B}^* e^{\check{A}^*(T-t)} \mathcal{P}^{-1}(T) z, \quad (3.115)$$

siehe z.B. [166, 188].

Analog erhält man mit Hilfe des dualen Systems (3.109) die positiv semidefinite Gramsche Beobachtbarkeitsmatrix.

Definition 3.21 (Endliche Gramsche Beobachtbarkeitsmatrix) Die endliche Gramsche Beobachtbarkeitsmatrix $\mathcal{Q}(T)$ für den Zeitpunkt $T < \infty$ des linearen Systems (3.12) ist definiert durch

$$\mathcal{Q}(T) = \int_0^T e^{\check{A}^* t} C^* C e^{\check{A} t} dt. \quad (3.116)$$

Ein Matrix-Paar (C, \check{A}) ist genau dann im Intervall $[0, T]$ beobachtbar, wenn $\mathcal{Q}(T) \in \mathbb{K}^{2n \times 2n}$ regulär und damit positiv definit ist. Mit Hilfe der Gramschen Beobachtbarkeitsmatrix $\mathcal{Q}(T)$ kann ermittelt werden, in welchem Maße die einzelnen Zustände zur beobachtbaren Energie beitragen. Für die hier ausschließlich betrachteten linearen zeitinvarianten Systeme gilt nach [10] außerdem, dass Steuer- und Beobachtbarkeit genau dann vorliegt, wenn $\mathcal{P}(T) > \mathbf{0}$ bzw. $\mathcal{Q}(T) > \mathbf{0}$ für ein beliebiges T gezeigt werden kann.

Betrachtet man ein stabiles System der Form (3.12), dann sind die Gramsche Steuerbarkeits- und Beobachtbarkeitsmatrix $\mathcal{P}(T)$ bzw. $\mathcal{Q}(T)$ auch für $T \rightarrow \infty$ definiert. Die unendlichen Gramschen Matrizen

$$P := \mathcal{P}(\infty), \quad (3.117)$$

$$Q := \mathcal{Q}(\infty), \quad (3.118)$$

gehören zu den wichtigsten Matrizen in der LTI-Systemtheorie. Diese Matrizen sind von besonderer Bedeutung, da sich diese für asymptotisch stabile \check{A} mit Hilfe der Lyapunov-Gleichungen berechnen lassen [10, 188].

Satz 3.6 Gegeben sei das (asymptotisch) stabile System (3.12). Die zugehörige unendlichen Gramschen Matrizen P und Q erfüllen dann die Lyapunov-Gleichungen

$$\check{A} \cdot P + P \cdot \check{A}^* + \check{B} \cdot \check{B}^* = \mathbf{0}, \quad (3.119)$$

$$\check{A}^* \cdot Q + Q \cdot \check{A} + C^* \cdot C = \mathbf{0}. \quad (3.120)$$

Liegt das System in der Form (3.4) vor, verwendet man alternativ die verallgemeinerten Lyapunov-Gleichungen

$$A \cdot P_G \cdot E^* + E \cdot P_G \cdot A^* + B \cdot B^* = \mathbf{0}, \quad (3.121)$$

$$A^* \cdot Q_G \cdot E + E^* \cdot Q_G \cdot A + C^* \cdot C = \mathbf{0}. \quad (3.122)$$

Dabei gilt der Zusammenhang:

$$P = P_G, \quad (3.123)$$

$$Q = E^* \cdot Q_G \cdot E. \quad (3.124)$$

Liegt ein System zweiter Ordnung vor, so kann mit Hilfe der Gleichungen (3.5) oder (3.6) ein System erster Ordnung der Form (3.4) definiert werden. Die Gramschen Steuerbarkeits- und

Beobachtbarkeitsmatrizen lassen sich dann unter Verwendung der verallgemeinerten Lyapunov-Gleichungen (3.121) und (3.122) bestimmen. Liegt darüber hinaus der Sonderfall eines symmetrischen Systems erster Ordnung (d.h. $\mathbf{C} = \mathbf{B}^*$) vor, so sind die beiden verallgemeinerten Lyapunov-Gleichungen identisch und es genügt eine der beiden Gleichungen zu lösen.

Die Hankel-Matrix (3.14) kann als Produkt der unendlichen Beobachtbarkeitsmatrix $\mathcal{O}(\mathbf{C}, \check{\mathbf{A}})$ (3.107) und der unendlichen Steuerbarkeitsmatrix $\mathcal{R}(\check{\mathbf{A}}, \check{\mathbf{B}})$ (3.102) zerlegt werden. Für ein System der Form (3.12) erhält man die Zerlegung:

$$\mathcal{H} = \begin{bmatrix} \mathbf{C}\check{\mathbf{B}} & \mathbf{C}\check{\mathbf{A}}\check{\mathbf{B}} & \mathbf{C}\check{\mathbf{A}}^2\check{\mathbf{B}} & \dots \\ \mathbf{C}\check{\mathbf{A}}\check{\mathbf{B}} & \mathbf{C}\check{\mathbf{A}}^2\check{\mathbf{B}} & \mathbf{C}\check{\mathbf{A}}^3\check{\mathbf{B}} & \dots \\ \mathbf{C}\check{\mathbf{A}}^2\check{\mathbf{B}} & \mathbf{C}\check{\mathbf{A}}^3\check{\mathbf{B}} & \mathbf{C}\check{\mathbf{A}}^4\check{\mathbf{B}} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad (3.125)$$

$$= \underbrace{\begin{bmatrix} \mathbf{C} \\ \mathbf{C}\check{\mathbf{A}} \\ \mathbf{C}\check{\mathbf{A}}^2 \\ \mathbf{C}\check{\mathbf{A}}^3 \\ \vdots \end{bmatrix}}_{\mathcal{O}(\mathbf{C}, \check{\mathbf{A}})} \underbrace{\begin{bmatrix} \check{\mathbf{B}} & \check{\mathbf{A}}\check{\mathbf{B}} & \check{\mathbf{A}}^2\check{\mathbf{B}} & \check{\mathbf{A}}^3\check{\mathbf{B}} & \dots \end{bmatrix}}_{\mathcal{R}(\check{\mathbf{A}}, \check{\mathbf{B}})}. \quad (3.126)$$

Die Matrizen $\mathcal{O}(\mathbf{C}, \check{\mathbf{A}})$ und $\mathcal{R}(\check{\mathbf{A}}, \check{\mathbf{B}})$ können durch die endlichen Matrizen $\mathcal{O}_{2n}(\mathbf{C}, \check{\mathbf{A}})$ (3.108) und $\mathcal{R}_{2n}(\check{\mathbf{A}}, \check{\mathbf{B}})$ (3.103) ausgetauscht werden. Da die Matrizen $\mathcal{O}_{2n}(\mathbf{C}, \check{\mathbf{A}})$ und $\mathcal{R}_{2n}(\check{\mathbf{A}}, \check{\mathbf{B}})$ jeweils maximal den Rang $2n$ besitzen, hat die Hankel-Matrix (3.95) ebenfalls maximal den Rang $2n$.

3.4.4 Hankel-Singulärwerte

An dieser Stelle wird der sogenannte Hankel-Operator definiert, der in der Systemtheorie und bei einigen Reduktionsverfahren von Bedeutung ist.

Definition 3.22 (Hankel-Operator) *Es sei ein stabiles, minimal realisiertes LTI-System gegeben. Betrachtet man den Eingang $\mathbf{u} : (-\infty, \infty) \rightarrow \mathbb{K}^m$ mit $\mathbf{u}(t) = 0$ für $t \geq 0$ und den Ausgang $\mathbf{y}(t)$ für $t \geq 0$, dann erhält man*

$$\mathbf{y}(t) = \int_{-\infty}^0 \mathbf{h}(t - \tau) \mathbf{u}(\tau) d\tau, \quad \text{für } t \geq 0, \quad (3.127)$$

wobei $\mathbf{h}(t)$ die Impulsantwort (3.40) des Systems bezeichnet. Der Operator

$$\mathcal{H} : \mathcal{L}_2(\mathbb{R}_-) \rightarrow \mathcal{L}_2(\mathbb{R}_+), \quad \mathbf{u}^- \mapsto \mathbf{y}^+, \quad \text{für } t \geq 0, \quad (3.128)$$

wobei

$$\mathbf{y}^+(t) = \mathcal{H}(\mathbf{u}^-)(t) = \int_{-\infty}^0 \mathbf{h}(t - \tau) \mathbf{u}^-(\tau) d\tau, \quad \text{für } t \geq 0, \quad (3.129)$$

welcher die zurückliegenden Eingangszustände $\mathbf{u}^- : (-\infty, 0] \rightarrow \mathbb{R}^m$ auf zukünftige Ausgänge $\mathbf{y}^+ : [0, \infty) \rightarrow \mathbb{R}^q$ abbildet, wird Hankel-Operator des Systems genannt.

In Matrixschreibweise läßt sich der Operator so schreiben:

$$\begin{bmatrix} \mathbf{y}_0 \\ \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{H}_0 & \mathbf{H}_1 & \mathbf{H}_2 & \cdots \\ \mathbf{H}_1 & \mathbf{H}_2 & \mathbf{H}_3 & \cdots \\ \mathbf{H}_2 & \mathbf{H}_3 & \mathbf{H}_4 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_{\mathcal{H}} \cdot \begin{bmatrix} \mathbf{u}_{-1} \\ \mathbf{u}_{-2} \\ \mathbf{u}_{-3} \\ \vdots \end{bmatrix} \quad (3.130)$$

wobei \mathbf{H}_k , $k = 0, 1, \dots, 2n$ die Markov-Parameter (3.89) bzw. (3.92) sind. Es ist somit ersichtlich, dass \mathcal{H} die selbe Matrix wie die Hankel-Matrix (3.95) ist.

Definition 3.23 (Hankel-Singulärwerte) Für ein stabiles LTI-System werden die positiven Wurzeln der Eigenwerte des Produktes der Gramschen Steuerbarkeitsmatrix (3.117) und der Gramschen Beobachtbarkeitsmatrix (3.118)

$$\sigma_k(\mathcal{H}) = \sqrt{\lambda_k(\mathcal{P}\mathcal{Q})}, \quad k = 1, \dots, 2n \quad (3.131)$$

Hankel-Singulärwerte genannt.

3.5 Zeit- und Frequenzbereichsräume und Normen

An dieser Stelle sollen die in dieser Arbeit verwendeten Signal- und Systemnormen eingeführt werden. Mit Hilfe der definierten Normen werden daraufhin Funktionsräume definiert, die in der Systemtheorie von großer Bedeutung sind. Hier können nur elementare Grundlagen dargestellt werden, die zum Verständnis der vorliegenden Arbeit notwendig sind. Detailliertere Informationen sind z.B. in [10, 53, 92, 94, 110, 148, 149] zu finden.

3.5.1 Signalknormen

3.5.1.1 Normen und Räume des Zeitbereichs

Eine Möglichkeit, um die Norm von Signalen im Zeitbereich zu berechnen, besteht in der Verwendung der sogenannten p -Normen. Nach [188] erhält man diese für eine stetige Funktion $f: \mathcal{I} \rightarrow \mathbb{R}$ auf $\mathcal{I} = \mathbb{R}$, $\mathcal{I} = \mathbb{R}_+$, $\mathcal{I} = \mathbb{R}_-$ oder einem Intervall $\mathcal{I} \subset \mathbb{R}$ mit

$$\|f\|_p = \begin{cases} \left(\int_{t \in \mathcal{I}} |f(t)|^p dt \right)^{\frac{1}{p}}, & \text{für } 1 \leq p < \infty \\ \text{ess sup}_{t \in \mathcal{I}} |f(t)|, & \text{für } p = \infty, \end{cases} \quad (3.132)$$

wobei $\text{ess sup}_{t \in \mathcal{I}} |f(t)|$ das Supremum von $f(t)$ fast überall, d. h. bis auf eine Menge mit Maß 0 bezeichnet. Werden matrix- oder vektorwertige Funktionen $\mathbf{f}(t)$ betrachtet, so können entsprechende Normen dadurch definiert werden, dass der Betrag $|\cdot|$ in (3.132) durch eine geeignete Matrix- oder Vektornorm ersetzt wird.

Definition 3.24 (p -Norm) Sei $\mathcal{I} = \mathbb{R}, \mathcal{I} = \mathbb{R}_+, \mathcal{I} = \mathbb{R}_-$ oder ein Intervall $\mathcal{I} \subset \mathbb{R}$ und $\mathbf{f} : \mathcal{I} \rightarrow \mathbb{R}^n$ eine stetige vektorwertige Funktion. Für $1 \leq p \leq \infty$ wird mit

$$\|\mathbf{f}\|_p = \begin{cases} \left(\int_{t \in \mathcal{I}} \|\mathbf{f}(t)\|_p^p dt \right)^{\frac{1}{p}}, & \text{für } 1 \leq p < \infty \\ \operatorname{ess\,sup}_{t \in \mathcal{I}} \|\mathbf{f}(t)\|_p, & \text{für } p = \infty \end{cases} \quad (3.133)$$

die p -Norm definiert. Dabei bezeichnet $\|\mathbf{f}(t)\|_p$ die p -Norm (2.26) von $\mathbf{f}(t) \in \mathbb{R}^n$.

Anmerkung 3.9 Die Aussage $\|\mathbf{f}\|_\infty = \gamma$ bedeutet, dass γ der kleinste Wert ist, der von $\mathbf{f}(t)$ auf dem Intervall \mathcal{I} nur auf Mengen vom Maß Null überschritten wird, für weitergehende Erläuterungen sei auf [7] verwiesen.

Mit Hilfe dieser Normen lassen sich die zugehörigen Banachräume der Lebesgue-integrierbaren Funktionen definieren. Diese Räume werden oft auch Zeitbereich genannt, da die Elemente dieser Räume Funktionen einer reellen Variablen sind, bei der es sich in physikalischen Anwendungen oft um die Zeit handelt.

Definition 3.25 (Banachraum der Lebesgue-integrierbaren Funktionen)

Sei $\mathcal{I} = \mathbb{R}, \mathcal{I} = \mathbb{R}_+, \mathcal{I} = \mathbb{R}_-$ oder einem Intervall $\mathcal{I} \subset \mathbb{R}$. Für $1 \leq p \leq \infty$ wird mit

$$\mathcal{L}_p^n(\mathcal{I}) = \{\mathbf{f} : \mathcal{I} \rightarrow \mathbb{R}^n : \|\mathbf{f}\|_p < \infty\}, \quad \text{für } 1 \leq p \leq \infty \quad (3.134)$$

der Banachraum der Lebesgue-integrierbaren Funktionen definiert, wobei $\|\mathbf{f}\|_p$ die p -Norm einer vektorwertigen Funktion \mathbf{f} bezeichnet.

3.5.1.2 Normen und Räume des Frequenzbereichs

Um Analysen im Frequenzbereich durchführen zu können, werden Normen von Funktionen einer komplexen Variablen benötigt.

Dazu werden matrixwertige Funktionen $\mathbf{F} : \mathbb{C} \rightarrow \mathbb{C}^{q \times m}$ betrachtet, die in der offenen rechten Halbebene $\mathbb{C}_+ \subset \mathbb{C}$ ($s = \beta + i\omega \in \mathbb{C}, \beta > 0$) der komplexen Ebene analytisch sind, wobei analytisch bedeutet, dass die Funktion in jedem Punkt von \mathbb{C}_+ in eine Taylorreihe entwickelt werden kann.

Definition 3.26 (\mathcal{H}_p -Normen) Sei $\mathbf{F} : \mathbb{C} \rightarrow \mathbb{C}^{q \times m}$ eine $q \times m$ komplexwertige Funktion, die in \mathbb{C}_+ analytisch ist. Für $1 \leq p \leq \infty$ wird mit

$$\|\mathbf{F}\|_{\mathcal{H}_p} = \begin{cases} \left(\frac{1}{2\pi} \sup_{\beta > 0} \int_{-\infty}^{\infty} \|\mathbf{F}(\beta + i\omega)\|_{S,p}^p d\omega \right)^{\frac{1}{p}}, & \text{für } 1 \leq p < \infty \\ \sup_{s \in \mathbb{C}_+} \|\mathbf{F}(s)\|_{S,p}, & \text{für } p = \infty \end{cases} \quad (3.135)$$

die \mathcal{H}_p -Norm definiert.

Dabei bezeichnet $\|\mathbf{F}(s)\|_{S,p}$ die Schatten- p -Norm (2.34) von \mathbf{F} an der Stelle $s = \beta + i\omega \in \mathbb{C}$. Mit Hilfe dieser Normen lassen sich die sogenannten Hardy-Räume (\mathcal{H}_p -Räume) definieren.

Definition 3.27 (Hardy-Räume (\mathcal{H}_p)) Für $1 \leq p \leq \infty$ wird mit $\mathcal{H}_p^{q \times m}(\mathbb{C}_+)$ der Raum aller in \mathbb{C}_+ analytischen Funktionen $\mathbf{F} : \mathbb{C} \rightarrow \mathbb{C}^{q \times m}$ mit $\|\mathbf{F}\|_{\mathcal{H}_p} < \infty$ bezeichnet. D.h.

$$\mathcal{H}_p^{q \times m}(\mathbb{C}_+) = \{\mathbf{F} : \mathbb{C} \rightarrow \mathbb{C}^{q \times m} : \|\mathbf{F}\|_{\mathcal{H}_p} < \infty\}, \quad \text{für } 1 \leq p \leq \infty. \quad (3.136)$$

Anmerkung 3.10 Die Hardy-Räume $\mathcal{H}_p^{q \times m}(\mathbb{C}_-)$ aller in \mathbb{C}_- analytischen Funktionen $\mathbf{F} : \mathbb{C} \rightarrow \mathbb{C}^{q \times m}$ können analog zu den Definition 3.26 und 3.27 definiert werden.

In dieser Arbeit werden nur Normen mit $p = 2$ und $p = \infty$ verwendet. In diesen Fällen ergeben sich mit (2.34) und (2.35) für die \mathcal{H}_2 - und die \mathcal{H}_∞ -Norm vereinfachte Schreibweisen. Die \mathcal{H}_2 -Norm ist damit gegeben durch:

$$\begin{aligned} \|\mathbf{F}\|_{\mathcal{H}_2} &= \left(\frac{1}{2\pi} \sup_{\beta > 0} \int_{-\infty}^{\infty} \text{spur}(\mathbf{F}^*(\beta - i\omega) \cdot \mathbf{F}(\beta + i\omega)) d\omega \right)^{\frac{1}{2}} \\ &= \left(\frac{1}{2\pi} \sup_{\beta > 0} \int_{-\infty}^{\infty} \text{spur}(\mathbf{F}(\beta + i\omega) \cdot \mathbf{F}^*(\beta - i\omega)) d\omega \right)^{\frac{1}{2}} \\ &= \left(\frac{1}{2\pi} \sup_{x > 0} \int_{-\infty}^{\infty} \|\mathbf{F}(\beta + i\omega)\|_F^2 d\omega \right)^{\frac{1}{2}}. \end{aligned} \quad (3.137)$$

Die \mathcal{H}_∞ -Norm ist damit gegeben durch:

$$\|\mathbf{F}\|_{\mathcal{H}_\infty} = \sup_{s \in \mathbb{C}_+} \sigma_{\max}(\mathbf{F}(s)). \quad (3.138)$$

Der \mathcal{H}_∞ -Raum beinhaltet somit alle in \mathbb{C}_+ analytischen und beschränkten matrixwertigen Funktionen \mathbf{F} [188].

Die Bestimmung der Suprema in den obigen Gleichungen können vereinfacht werden, indem das *Maximum-Modulus-Theorem* angewendet wird. Dieses besagt, dass eine Funktion f , die auf einem Bereich $D \in \mathbb{C}$ und auf dem Rand ∂D stetig sowie auf D analytisch ist, ihr Maximum auf dem Rand ∂D von D annimmt. Wendet man dieses Theorem auf die Gleichungen (3.137) und (3.138) an erhält man

$$\|\mathbf{F}\|_{\mathcal{H}_2} = \left(\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{spur}(\mathbf{F}^*(-i\omega) \cdot \mathbf{F}(i\omega)) d\omega \right)^{\frac{1}{2}} \quad (3.139)$$

und

$$\|\mathbf{F}\|_{\mathcal{H}_\infty} = \sup_{\omega \in \mathbb{R}} \sigma_{\max}(\mathbf{F}(i\omega)). \quad (3.140)$$

Hat die Funktion $\mathbf{F} : \mathbb{C} \rightarrow \mathbb{C}^{q \times m}$ keine Singularitäten (Pole) auf der imaginären Achse und ist weder in der offenen rechten Halbebene \mathbb{C}_+ noch der offenen linken Halbebene \mathbb{C}_- der komplexen Ebene analytisch, dann ist die \mathcal{H}_p -Norm nicht definiert. In diesem Fall können die folgend definierten \mathcal{L}_p -Normen für den Frequenzbereich verwendet werden.

Definition 3.28 (\mathcal{L}_p -Normen für den Frequenzbereich) Sei $\mathbf{F} : \mathbb{C} \rightarrow \mathbb{C}^{q \times m}$ eine $q \times m$ komplexwertige Funktion. Für $1 \leq p \leq \infty$ wird mit

$$\|\mathbf{F}\|_{\mathcal{L}_p} = \begin{cases} \left(\sup_{\omega \in \mathbb{R}} \int_{-\infty}^{\infty} \|\mathbf{F}(i\omega)\|_p^p d\omega \right)^{\frac{1}{p}}, & \text{für } 1 \leq p < \infty \\ \sup_{\omega \in \mathbb{R}} \sigma_{\max}(\mathbf{F}(i\omega)), & \text{für } p = \infty \end{cases} \quad (3.141)$$

die \mathcal{L}_p -Norm für den Frequenzbereich definiert.

Mit Hilfe dieser Normen lassen sich die \mathcal{L}_p -Räume für den Frequenzbereich definieren.

Definition 3.29 (\mathcal{L}_p -Räume für den Frequenzbereich) Für $1 \leq p \leq \infty$ wird mit $\mathcal{L}_p(i\mathbb{R})$ der Raum aller $\mathbf{F} : \mathbb{C} \rightarrow \mathbb{C}^{q \times m}$ mit $\|\mathbf{F}\|_{\mathcal{L}_p} < \infty$ bezeichnet. D.h.

$$\mathcal{L}_p(i\mathbb{R}) = \{\mathbf{F} : \mathbb{C} \rightarrow \mathbb{C}^{q \times m} : \|\mathbf{F}\|_{\mathcal{L}_p} < \infty\}, \quad \text{für } 1 \leq p \leq \infty. \quad (3.142)$$

3.5.2 Systemnormen

Zur Definition von Systemnormen werden LTI Systeme der Form (3.12) betrachtet. Eine wichtige Norm zur Analyse von Systemen ist die Hankel-Norm, die mit Hilfe der Hankel-Singulärwerte des Systems bestimmt werden kann.

Definition 3.30 (Hankel-Norm) Die Hankel-Norm eines stabilen LTI-Systems $\check{\Sigma}_{1o}$ ist durch

$$\|\check{\Sigma}_{1o}\|_H = \sup_{u \in \mathcal{L}_2(-\infty, 0)} \frac{\|y\|_2}{\|u\|_2} \quad (3.143)$$

definiert, wobei $\mathcal{L}_2(-\infty, 0)$ den Banachraum der auf dem Intervall $[-\infty, 0]$ zweifach Lebesgue-integrierbaren Funktionen bezeichnet.

Die Hankel-Norm entspricht dem betragsgrößten Hankel-Singulärwert des Systems $\check{\Sigma}_{1o}$ [76]. Daraus lässt sich mit (3.131) schließen, dass gilt

$$\|\check{\Sigma}_{1o}\|_H = \sigma_{\max} = \max_{1 \leq k \leq 2n} \{\sigma_k(\mathcal{H})\} = \max_{1 \leq k \leq 2n} \{\sqrt{\lambda(\mathcal{P}\mathcal{Q})}\}. \quad (3.144)$$

Weiterhin gilt für die Hankel-Norm der Übertragungsfunktion $\mathbf{H}(s) = \mathbf{C}(s\mathbf{I} - \check{\mathbf{A}})^{-1}\check{\mathbf{B}}$:

$$\|\check{\Sigma}_{1o}\|_H = \|\mathbf{H}(s)\|_H = \sigma_{\max}. \quad (3.145)$$

Eine weitere Norm zur Analyse von Systemen ist die \mathcal{H}_2 -Norm, die als die \mathcal{L}_2 -Norm der Impulsantwort im Zeitbereich definiert ist:

$$\|\check{\Sigma}_{1o}\|_{\mathcal{H}_2} = \|\mathbf{h}\|_{\mathcal{L}_2}. \quad (3.146)$$

Diese Norm ist nur für stabile Systeme beschränkt [10]. In diesem Fall gilt mit (3.139):

$$\|\check{\Sigma}_{1o}\|_{\mathcal{H}_2}^2 = \int_0^{\infty} \text{spur}((\mathbf{h}(t))^* \cdot \mathbf{h}(t)) dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{spur}((\mathbf{H}(-i\omega))^* \cdot \mathbf{H}(i\omega)) d\omega. \quad (3.147)$$

Die zweite Gleichung folgt dabei durch Verwendung des Satzes von Parseval. Mit der Gleichung (3.116) für $T = \infty$ erhält man:

$$\|\check{\Sigma}_{1o}\|_{\mathcal{H}_2}^2 = \int_0^\infty \text{spur}\left(\mathbf{B}e^{\check{\mathbf{A}}^*t}\mathbf{C}^*\mathbf{C}e^{\check{\mathbf{A}}t}\mathbf{B}\right)dt = \text{spur}(\check{\mathbf{B}}^*\mathbf{Q}\check{\mathbf{B}}). \quad (3.148)$$

Da gilt $\text{spur}(\mathbf{h}^*(t)\mathbf{h}(t)) = \text{spur}(\mathbf{h}(t)\mathbf{h}^*(t))$, folgt mit (3.114) für $T = \infty$, dass ebenfalls die Gleichung

$$\|\check{\Sigma}_{1o}\|_{\mathcal{H}_2}^2 = \text{spur}(\mathbf{C}\mathbf{P}\mathbf{C}^*) \quad (3.149)$$

gilt.

Definition 3.31 (\mathcal{H}_2 -Norm) Die \mathcal{H}_2 -Norm eines LTI-Systems $\check{\Sigma}_{1o}$ ist durch

$$\|\check{\Sigma}_{1o}\|_{\mathcal{H}_2} = \sqrt{\text{spur}(\check{\mathbf{B}}^*\mathbf{Q}\check{\mathbf{B}})} = \sqrt{\text{spur}(\mathbf{C}\mathbf{P}\mathbf{C}^*)} \quad (3.150)$$

definiert, wobei \mathbf{P} die unendliche Gramsche Steuerbarkeitsmatrix (3.117) und \mathbf{Q} die unendliche Gramsche Beobachtbarkeitsmatrix (3.118) des Systems sind.

Da nach [36] die Frobenius-Norm, von der sich die \mathcal{H}_2 -Norm ableitet, keine induzierte Norm ist, aber in [37] gezeigt wird, dass $\sqrt{\lambda_{\max}(\mathbf{C}\mathbf{P}\mathbf{C}^*)}$ eine induzierte Norm ist, folgt, dass die \mathcal{H}_2 -Norm nur im Fall eines SISO-Systems eine induzierte Norm ist [10].

Kapitel 4

Grundlagen der Strukturdynamik

In diesem Kapitel sollen zunächst die zum Verständnis des weiteren Vorgehens wichtigen Grundlagen der Strukturdynamik dargestellt werden. Die Darstellungen dieses Kapitels orientieren sich dabei an den Darstellungen in [75].

Eine Struktur ist als ein lineares System definiert, welches folgende Eigenschaften aufweist:

- Es ist endlich dimensional.
- Es ist steuerbar und beobachtbar.
- Seine Pole sind komplex mit kleinem Realteil.
- Die Pole treten nicht gehäuft auf.

Des weiteren zeichnen sich mechanische Strukturen zusätzlich durch die folgenden Eigenschaften aus:

- Das dynamische Verhalten einer mechanischen Struktur kann durch unabhängige Koordinaten beschrieben werden, die Moden genannt werden.
- Vorhandene Pole treten konjugiert komplex auf. Dabei ist der Realteil typischerweise klein.
- Die Impulsantwort einer mechanischen Struktur besteht aus harmonischen Komponenten, die mit komplexen Polen in Verbindung stehen.
- Jedes konjugiert komplexe Polpaar repräsentiert eine Strukturmode.
- Der Realteil eines Poles repräsentiert die Dämpfung einer Mode.
- Der Absolutbetrag eines Poles repräsentiert die Eigenfrequenz einer Mode.

Strukturmodelle können entweder mit Hilfe physikalischer Gesetze, wie z. B. mit den Newtonschen Bewegungsgesetzen, den Bewegungsgleichungen nach Lagrange, mit dem Prinzip nach D'Alembert [130] oder mit der Methode der Finiten Elemente (FE) [189] gewonnen werden. Dabei können die Modelle entweder im Zeitbereich, durch ein System von Differenzialgleichungen (DGL), oder im Frequenzbereich, durch eine Übertragungsfunktion, dargestellt werden.

Für die strukturmechanischen Berechnungen werden in dieser Arbeit Finite-Elemente-Modelle verwendet. In Abschnitt 7.3 wird näher dargestellt, dass hierbei auf einen umfangreichen Kenntnisstand aufgebaut werden kann.

Im ersten Abschnitt 4.1 dieses Kapitels erfolgt zunächst die Beschreibung der in dieser Arbeit verwendeten Darstellung von Strukturmodellen durch lineare zeitinvariante Differenzialgleichungssysteme. Darauf erfolgt im Abschnitt 4.2 die Beschreibung der Methode der Modaltransformation dieser Systeme. Abschnitt 4.3 widmet sich dem Problem der Dämpfung eines mechanischen Systems. Im abschließenden Abschnitt 4.4 dieses Kapitels erfolgt die spezielle Beschreibung von Strukturmodellen durch ihre Übertragungsfunktionen.

4.1 Strukturmodelle

Das Verhalten mechanischer Systeme wird üblicherweise mit Differenzialgleichungen zweiter Ordnung beschrieben. In dieser Arbeit wird dabei der wichtige Spezialfall einer Beschreibung mit Hilfe eines Systems linearer zeitinvarianter Differenzialgleichungen zweiter Ordnung betrachtet. Bei diesen Differenzialgleichungssystemen dienen die Freiheitsgrade des zugrunde liegenden Strukturmodells zur Beschreibung der Dynamik des mechanischen Systems.

Die Modellierung einer mechanischen Struktur erfolgt üblicherweise mit der FEM. Eine ausführlichere Beschreibung der Anwendung der Methode der finite Elemente zur Modellierung von Werkzeugmaschinen erfolgt in Abschnitt 7.3.2. Die FEM liefert, unter der Annahme eines linearen Verhaltens der Störbewegungen $\mathbf{x}(t)$, ein Differenzialgleichungssystem zweiter Ordnung (Bewegungsgleichungssystem) der Form:

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) &= \mathbf{F}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}_p \mathbf{x}(t) + \mathbf{C}_v \dot{\mathbf{x}}(t), \end{aligned} \quad (4.1)$$

wobei $\mathbf{M}, \mathbf{K} \in \mathbb{R}^{n \times n}$, $\mathbf{F} \in \mathbb{R}^{n \times m}$ und $\mathbf{C}_p, \mathbf{C}_v \in \mathbb{R}^{q \times n}$. \mathbf{M} stellt in dieser Darstellung die Massenmatrix und \mathbf{K} die Steifigkeitsmatrix des Systems dar. Die Matrizen \mathbf{F} sowie \mathbf{C}_p und \mathbf{C}_v werden Eingangs- bzw. Ausgangsmatrizen genannt. Man nennt $\mathbf{x}(t) \in \mathbb{R}^n$ die Position oder den Zustand des Systems, $\mathbf{u}(t) \in \mathbb{R}^m$ den Eingang und $\mathbf{y}(t) \in \mathbb{R}^q$ den Ausgang des Systems.

Bei der Modellierung der Werkzeugmaschinenstruktur mit finiten Elementen wird zunächst auf die Modellierung der Dämpfung verzichtet. Diese kann in Form einer Dämpfungsmatrix $\mathbf{D} \in \mathbb{R}^{n \times n}$ hinzugefügt werden. In Abschnitt 4.3 wird das Problem der Dämpfung von Strukturmodellen durch die Wahl einer geeigneten Dämpfungsmatrix näher betrachtet. Es ergibt sich somit die Beschreibung des gedämpften Gesamtsystems zu:

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{D}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) &= \mathbf{F}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}_p \mathbf{x}(t) + \mathbf{C}_v \dot{\mathbf{x}}(t). \end{aligned} \quad (4.2)$$

Werden bei der Modellierung der mechanischen Struktur mit der FEM nicht-massebehaftete Federelemente zur Verbindung von Maschinenteilen und Schalenelemente eingesetzt, ist die Massenmatrix im Allgemeinen singulär. Man erhält in diesem Fall ein System von Algebra-Differenzialgleichungen (DAE) zweiter Ordnung der Form (4.2).

Unter Vernachlässigung gyroskopischer Kräfte (Kreiseffekte) und nichtkonservativer Lagekräfte (z.B. Reib- oder zirkulatorische Kräfte) sind die Koeffizientenmatrizen symmetrisch mit

$$\mathbf{M} = \mathbf{M}^T > \mathbf{0}, \quad \mathbf{D} = \mathbf{D}^T \geq \mathbf{0}, \quad \text{und} \quad \mathbf{K} = \mathbf{K}^T \geq \mathbf{0}. \quad (4.3)$$

Im Fall des häufig verwendeten konzentrierten Massenansatzes ist die Massenmatrix sogar diagonal.

Definition 4.1 *Ein System der Form (4.1) heißt*

- *konservativ, wenn für die Dämpfungsmatrix $\mathbf{D} \equiv \mathbf{0}$ gilt.*
- *dissipativ, wenn für die Dämpfungsmatrix $\mathbf{D} \neq \mathbf{0}$ gilt.*

Das erhaltene System beschreibt das Verhalten der mechanischen Struktur. Es können nun Kräfte und Momente auf das Modell der Struktur gegeben werden, um Analysen bezüglich des Verhaltens der Struktur anstellen zu können.

4.2 Die Modaltransformation

Die Dimension und die Bandbreite¹ der Systemmatrizen der Bewegungsgleichungen werden durch den Diskretisierungsgrad und die Topologie des erzeugten Strukturmodells bestimmt.

Da die Bandbreite den Kopplungsgrad der Gleichungen bestimmt und der Rechenaufwand mit der Zahl der Kopplungen zunimmt, wird häufig eine Reduktion der Bandbreite angestrebt. Während durch Permutation der Zeilen- und Spalten der Systemmatrizen lediglich eine Minimierung der Bandbreite erreichbar ist, kann, wenn die Bedingungen (4.3) vorliegen, durch Übergang auf generalisierte Koordinaten eine Diagonalstruktur der Matrizen erzielt werden, wodurch das Gleichungssystem vollständig entkoppelt wird [145].

Hierzu wird ein Ähnlichkeitsoperator Φ gesucht, mit dessen Hilfe die Knotenverschiebungen \mathbf{x} in Abhängigkeit der generalisierten Verschiebungen $\mathbf{q}(t)$ angegeben werden können:

$$\mathbf{x}(t) = \Phi \mathbf{q}(t). \quad (4.4)$$

Zur Berechnung eines Operators Φ wird die homogene Systemgleichung

$$\mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{D}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) = \mathbf{0} \quad (4.5)$$

des Systems zweiter Ordnung der Form (4.2) betrachtet. Einsetzen des Ansatzes $\mathbf{x} = e^{\lambda_i t} \phi_i$ mit $\lambda_i \in \mathbb{C}$ und $\phi_i \in \mathbb{C}^n$ führt auf das quadratische Eigenwertproblem

$$(\lambda_i^2 \mathbf{M} + \lambda_i \mathbf{D} + \mathbf{K}) \cdot \phi_i = \mathbf{0}, \quad (4.6)$$

siehe [3]. Sind die Eigenschaften (4.3) erfüllt, dann besitzt das quadratische Eigenwertproblem (4.6) $2n$ reelle oder paarweise komplex konjugierte Eigenwerte λ_i , $i = 1, \dots, n$. Jeder Rechtseigenvektor ϕ_i zu λ_i ist dann auch gleichzeitig ein Linkseigenvektor $\bar{\phi}_i$ zum konjugiert komplexen Eigenvektor $\bar{\lambda}_i$ mit

$$\bar{\phi}_i \cdot (\bar{\lambda}_i^2 \mathbf{M} + \bar{\lambda}_i \mathbf{D} + \mathbf{K}) = \mathbf{0}, \quad (4.7)$$

¹Als Bandbreite einer Matrix wird in der numerischen Mathematik die Anzahl der Hauptdiagonalen plus der Anzahl der Nebendiagonalen, deren Elementen ungleich Null sind, bezeichnet.

siehe [177]. Die Eigenvektoren sind dabei reell oder der Linkseigenvektor liegt als das komplex konjugierte des Rechtseigenvektors vor.

Besitzt das System keine oder eine proportionale Dämpfungsmatrix, so lässt sich anstelle der Gleichung (4.5) das Eigenwertproblem des konservativen Systems betrachten, da dessen Eigenvektoren mit denen des gedämpften Systems übereinstimmen. Beispielsweise lässt sich dazu die freie Schwingung des konservativen Systems betrachten. An Stelle der Gleichung (4.5) wird in diesem Fall die homogene Gleichung (4.1) des Systems

$$\mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) = \mathbf{0} \quad (4.8)$$

betrachtet. Einsetzen von $\mathbf{x} = e^{\lambda_i t} \boldsymbol{\phi}_i$ mit $\lambda_i \in \mathbb{C}$ und $\boldsymbol{\phi}_i \in \mathbb{C}^n$ in Gleichung (4.8) ergibt das Eigenwertproblem:

$$(\mathbf{K} + \omega_i^2 \mathbf{M}) \boldsymbol{\phi}_i = \mathbf{0}. \quad (4.9)$$

Bei der Gleichung (4.9) handelt es sich um ein verallgemeinertes Eigenwertproblem der Form (2.8), das in Abschnitt 2.1 bereits beschrieben wurde. Die nichttrivialen Lösungen dieses Eigenwertproblems sind durch die Eigenwerte ω_i^2 , $i = 1, \dots, n$ gegeben, wobei der Wert ω_i die i -te Eigenfrequenz genannt wird. Einsetzen der Werte ω_i in Gleichung (4.9) liefert zu jedem Eigenwert ω_i^2 den korrespondierenden Eigenvektor $\boldsymbol{\phi}_i$, $i = 1, \dots, n$, der die Gleichung erfüllt. Der i -te Vektor $\boldsymbol{\phi}_i$ wird i -te Eigenmode genannt. Die Eigenwerte λ_i sind, bei symmetrischer und positiv definiter Matrix \mathbf{M} und symmetrischer Matrix \mathbf{K} , negativ und reell und können als $-\lambda_i = \omega_i^2$ mit der Eigenfrequenz $\omega_i \in \mathbb{R}$ geschrieben werden.

Die Gleichung (4.9) gewährleistet zwar die Bestimmbarkeit linear unabhängiger Eigenvektoren, allerdings sind diese nicht eindeutig definiert und bedürfen der Normierung. Die Eigenvektoren bilden eine Basis, wobei deren Orthogonalität hinsichtlich der Entkopplung des Systems eine wichtige Eigenschaft darstellt. Es ist üblich, als weitere Bedingung für die Eindeutigkeit der Eigenvektoren die Massen-Orthonormalität einzuführen:

$$\boldsymbol{\phi}_i^T \mathbf{M} \boldsymbol{\phi}_j = \delta_{ij}, \quad (4.10)$$

wobei δ_{ij} das Kronecker-Symbol mit

$$\delta_{ij} = \begin{cases} 1, & \text{für } i = j \\ 0, & \text{für } i \neq j \end{cases} \quad (4.11)$$

darstellt. Durch diese Normierung beschreiben die generalisierten Verschiebungen $\mathbf{q}(t) = \boldsymbol{\Phi}^T \mathbf{x}(t)$ die Bewegungen von Einheitsmassen.

Die Eigenvektoren $\boldsymbol{\phi}_i$ können nun spaltenweise zur Eigenvektormatrix $\boldsymbol{\Phi}$ und die Eigenwerte ω_i^2 zur diagonalen Eigenwertmatrix $\boldsymbol{\Omega}^2$ zusammengefasst werden:

$$\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \dots, \boldsymbol{\phi}_n] \quad \text{und} \quad \boldsymbol{\Omega}^2 = \text{diag}(\omega_i^2). \quad (4.12)$$

Damit lautet die Lösung des Eigenwertproblems: (4.9)

$$\mathbf{K}\boldsymbol{\Phi} + \mathbf{M}\boldsymbol{\Phi}\boldsymbol{\Omega}^2 = \mathbf{0}. \quad (4.13)$$

Nach Multiplikation mit Φ^T von links, gilt wegen der Massen-Orthonormalität (4.10):

$$\Phi^T M \Phi = I \quad \text{und} \quad \Phi^T K \Phi = \Omega^2. \quad (4.14)$$

Die Gleichungen (4.14) machen deutlich, dass die Eigenvektormatrix Φ als Ähnlichkeitsoperator zur Diagonalisierung der Systemmatrizen M und K aus Gleichung (4.1) verwendet werden kann. Da die Eigenlösungen jeweils eine Schwingungsmode der Struktur beschreiben, bezeichnet man den Operator Φ als Modalmatrix oder Modalbasis und die Transformation (4.4) als Modaltransformation. Letztere beschreibt die Knotenverschiebung \mathbf{x} durch Linearkombination der Eigenvektoren ϕ_i mit den generalisierten (modalen) Verschiebungen $\mathbf{q}(t)$ als Linearfaktoren. Ein Eigenvektor ϕ_i stellt dabei die Schwingungsform einer Mode mit der Eigenkreisfrequenz ω_i dar.

Wendet man die so bestimmte Modaltransformation (4.4) auf das gekoppelte Gleichungssystem (4.2) an, so erhält man nach linksseitiger Multiplikation mit Φ^T die entkoppelten Bewegungsgleichungen in den modalen Koordinaten $\mathbf{q}(t)$:

$$\begin{aligned} \bar{M} \ddot{\mathbf{q}}(t) + \bar{D} \dot{\mathbf{q}}(t) + \bar{K} \mathbf{q}(t) &= \bar{\mathbf{F}} \mathbf{u}(t) \\ \mathbf{y}(t) &= \bar{\mathbf{C}}_p \mathbf{q}(t) + \bar{\mathbf{C}}_v \dot{\mathbf{q}}(t) \end{aligned} \quad (4.15)$$

mit

$$\begin{aligned} \bar{M} &= \Phi^T M \Phi = I, & \bar{D} &= \Phi^T D \Phi, & \bar{K} &= \Phi^T K \Phi = \Omega^2, \\ \bar{\mathbf{F}} &= \Phi^T \mathbf{F}, & \bar{\mathbf{C}}_p &= \mathbf{C}_p \Phi, & \bar{\mathbf{C}}_v &= \mathbf{C}_v \Phi. \end{aligned} \quad (4.16)$$

Die Darstellungsform (4.16) wird Modaldarstellung des Modells genannt.

Während die Diagonalform der modalen Massenmatrix \bar{M} und der modalen Steifigkeitsmatrix \bar{K} durch die Bestimmung der Modalmatrix Φ über das reelle Eigenwertproblem (4.9) des konservativen Systems (4.8) gewährleistet wird, gelingt die Diagonalisierung der Dämpfungsmatrix nur für den Sonderfall einer zur Massen- und Steifigkeitsmatrix proportionalen Dämpfung, siehe dazu Abschnitt 4.3.

4.3 Dämpfung

Grundsätzlich kann die Dämpfungsmatrix D in Gleichung (4.3), ebenso wie die Massen- und Steifigkeitsmatrizen, aus elementweisen Beiträgen zusammengesetzt werden, siehe dazu [18, 190]. Die Bestimmung physikalischer Dämpfungsparameter ist jedoch schwierig und mit großen Streuungen behaftet, da die Dämpfungsverhältnisse in mechanischen Systemen einer Vielzahl von Einflüssen unterliegen. Die Dämpfungsmatrix D kann daher im Allgemeinen und besonders bei FE-Modellen komplexer modellierter mechanischer Strukturen nicht in der gleichen Weise wie die Massen- und Steifigkeitsmatrix durch Elemente mit diskreten Dämpfungseigenschaften konstruiert werden. In der praktischen Anwendung muss die dissipierende Energie somit näherungsweise mit einem globalen Ansatz erfasst werden.

Ist die Dämpfungsmatrix \mathbf{D} positiv definit, so spricht man von vollständiger Dämpfung. Ist \mathbf{D} nur semidefinit, wird jedoch die Bedingung

$$\begin{aligned} \det \mathbf{K} &\neq \mathbf{0}, \\ \text{rang}([\mathbf{M}^{-1}\mathbf{D} \quad (\mathbf{M}^{-1}\mathbf{K})\mathbf{M}^{-1}\mathbf{D} \quad \dots \quad (\mathbf{M}^{-1}\mathbf{K})^{n-1}\mathbf{M}^{-1}\mathbf{D}]) &= n \end{aligned} \quad (4.17)$$

erfüllt, liegt durchdringende Dämpfung vor [117].

Anmerkung 4.1 Für strukturmechanische Probleme ist die Dämpfungsmatrix gewöhnlich symmetrisch und aus physikalischen Gründen auch positiv definit, siehe [3, 128].

Für Werkzeugmaschinen haben sich, da die Dämpfung typischerweise niedrig und im System verteilt ist, sowohl der modale als auch der proportionale Dämpfungsansatz als gute Näherungen erwiesen [56, 107, 175].

4.3.1 Modaler Dämpfungsansatz

Der modalen Dämpfung liegt die Annahme zu Grunde, dass jede Eigenfrequenz für sich gedämpft wird, wobei die Kopplung zwischen den Eigenfrequenzen infolge der Dämpfung vernachlässigt wird. Somit gilt

$$\Phi_i^T \cdot \mathbf{D} \cdot \Phi_j = \begin{cases} 0, & \text{für } i \neq j \\ 2\omega_i D_{L,i}, & \text{für } i = j \end{cases} \quad (4.18)$$

mit der i -ten Eigenmode Φ_i , der Eigenkreisfrequenz ω_i und dem Lehrschen Dämpfungsmaß $D_{L,i}$ der i -ten Eigenfrequenz des ungedämpften Systems [18].

In vielen Fällen, in welchen Erfahrung mit spezifischen mechanischen Strukturen vorhanden sind, sind die Lehrschen Dämpfungsmaße $D_{L,i}$ bekannt oder können geschätzt werden. Diese Lehrschen Dämpfungsmaße können in den entkoppelten Bewegungsgleichungen (4.15) berücksichtigt werden, indem sie in eine Lehr'sche Dämpfungsmatrix der Form

$$\mathbf{D}_L = \text{diag}(D_{L,i}) \quad (4.19)$$

eingebraucht werden.

Die für die modale Darstellung der Bewegungsgleichungen (4.15) geforderte Diagonalform der Dämpfungsmatrix $\bar{\mathbf{D}}$ in (4.16) wird somit mit

$$\bar{\mathbf{D}} = \Phi^T \mathbf{D} \Phi = 2\mathbf{\Omega} \mathbf{D}_L = \text{diag}(2\omega_i D_{L,i}) \quad (4.20)$$

unmittelbar erfüllt.

Wegen der Diagonalgestalt der Matrix $\mathbf{\Omega}$ kann das Gleichungssystem (4.16) auch als

$$\begin{aligned} \ddot{\mathbf{q}}(t) + \text{diag}(2\omega_i D_{L,i}) \dot{\mathbf{q}}(t) + \text{diag}(\omega_i^2) \mathbf{q}(t) &= \bar{\mathbf{F}} \mathbf{u}(t) \\ \mathbf{y}(t) &= \bar{\mathbf{C}}_p \mathbf{q}(t) + \bar{\mathbf{C}}_v \dot{\mathbf{q}}(t), \end{aligned} \quad (4.21)$$

mit $\bar{\mathbf{F}}$, $\bar{\mathbf{C}}_p$ und $\bar{\mathbf{C}}_v$ wie in (4.16), geschrieben werden.

4.3.2 Proportionaler Dämpfungsansatz

Allgemein lässt sich eine Dämpfungsmatrix durch eine Reihenentwicklung nach Caughey als

$$\mathbf{D} = \mathbf{M} \cdot \sum_{k=0}^{r-1} a_k (\mathbf{M}^{-1} \cdot \mathbf{K})^k \quad (4.22)$$

konstruieren [35]. Die Koeffizienten a_k können beispielsweise anhand von Messwerten festgelegt werden, siehe [18, 3].

Soll das Gleichungssystem ohne vorherige Reduktion der Dimension gelöst werden oder sind die Moden des Systems nicht bekannt, lässt sich aus der Gleichung (4.22) mit $r = 2$ der häufig verwendete Sonderfall der Rayleigh-Dämpfung, eines proportionalen Ansatzes aus einer Linearkombination der Massen- und Steifigkeitsmatrix mit Dämpfungsparametern $\alpha, \beta < 1$ verwenden:

$$\mathbf{D} = \alpha \mathbf{M} + \beta \mathbf{K}. \quad (4.23)$$

Die für die Darstellung der Bewegungsgleichungen (4.15) geforderte Diagonalform der Dämpfungsmatrix $\bar{\mathbf{D}}$ in (4.16) wird dann durch die Gleichung

$$\bar{\mathbf{D}} = \Phi^T (\alpha \mathbf{M} + \beta \mathbf{K}) \Phi = \alpha \mathbf{I} + \beta \mathbf{\Omega}^2 = \text{diag}(d_i), \quad (4.24)$$

wobei $d_i = \alpha + \beta \omega_i^2$ gilt, erfüllt.

Damit kann das Gleichungssystem (4.16) auch als

$$\begin{aligned} \ddot{\mathbf{q}}(t) + \text{diag}(\alpha + \beta \omega_i^2) \dot{\mathbf{q}}(t) + \text{diag}(\omega_i^2) \mathbf{q}(t) &= \bar{\mathbf{F}} \mathbf{u}(t) \\ \mathbf{y}(t) &= \bar{\mathbf{C}}_p \mathbf{q}(t) + \bar{\mathbf{C}}_v \dot{\mathbf{q}}(t), \end{aligned} \quad (4.25)$$

mit $\bar{\mathbf{F}}$, $\bar{\mathbf{C}}_p$ und $\bar{\mathbf{C}}_v$ wie in (4.16), geschrieben werden.

Durch Gleichsetzen von (4.20) und (4.24) erhält man eine Vorschrift zur Berechnung des i -ten Lehr'schen Dämpfungsmaßes $D_{L,i}$ für den Fall einer proportionalen Dämpfung des Systems:

$$\alpha + \beta \omega_i^2 = 2\omega_i D_{L,i} \Rightarrow D_{L,i} = \alpha \cdot \frac{1}{2\omega_i} + \beta \cdot \frac{\omega_i}{2}, \quad \text{für } i = 1, \dots, n. \quad (4.26)$$

4.4 Übertragungsfunktionen von Strukturmodellen

Um globale Aussagen über das dynamische Verhalten von mechanischen Strukturen treffen zu können, ist neben der Darstellung im Zeitbereich die Darstellung im Frequenzbereich üblich. Dazu werden die Übertragungsfunktionen zwischen den Orten der Krafteinleitung (Systemeingang) und des Abgriffs von Systemantworten (Systemausgang) berechnet [175].

Dazu wird das Übertragungsverhalten von $\mathbf{u}(s)$ nach $\mathbf{y}(s)$ an den Stelle s des Frequenzbereiches betrachtet:

$$\mathbf{Y}(s) = \mathbf{H}(s) \mathbf{U}(s), \quad (4.27)$$

wobei $\mathbf{U}(s)$ und $\mathbf{Y}(s)$ die Laplace-Transformierten des Eingangs $\mathbf{u}(t)$ und des Ausgangs $\mathbf{y}(t)$ des linearen Systems darstellen.

Wendet man die Laplace-Transformation auf die Gleichungen (4.2) eines Systems zweiter Ordnung um den Punkt $\mathbf{x}(0) = \mathbf{0}$ an, erhält man die Übertragungsfunktion:

$$\mathbf{H}(s) = (\mathbf{C}_p + s\mathbf{C}_v)(s^2\mathbf{M} + s\mathbf{D} + \mathbf{K})^{-1}\mathbf{F}, \quad (4.28)$$

siehe Gleichung (3.25) aus Abschnitt 3.2.1.

Anmerkung 4.2 Mit Hilfe der Anmerkung 3.3 aus Abschnitt 3.1 ergibt sich, dass die Übertragungsfunktion unter der Koordinaten-Transformation invariant ist, d. h. es gilt

$$(\tilde{\mathbf{C}}_p + s\tilde{\mathbf{C}}_v)(s^2\tilde{\mathbf{M}} + s\tilde{\mathbf{D}} + \tilde{\mathbf{K}})^{-1}\tilde{\mathbf{F}} = (\mathbf{C}_p + s\mathbf{C}_v)(s^2\mathbf{M} + s\mathbf{D} + \mathbf{K})^{-1}\mathbf{F}, \quad (4.29)$$

für $\tilde{\mathbf{M}}, \tilde{\mathbf{D}}, \tilde{\mathbf{K}}, \tilde{\mathbf{F}}, \tilde{\mathbf{C}}_p, \tilde{\mathbf{C}}_v$ wie in (3.10).

Mit Verwendung von (4.21) bzw. (4.25) in (4.28) kann die Übertragungsfunktion einer einzelnen Eigenmode wie in der folgenden Definition angegeben werden.

Definition 4.2 (Übertragungsfunktion einer einzelnen Eigenmode) Die Übertragungsfunktion der i -ten Eigenmode ist definiert durch

$$\mathbf{H}_i(s) = \frac{[\bar{\mathbf{C}}_p + s\bar{\mathbf{C}}_v]_{*i} \cdot [\bar{\mathbf{F}}]_{i*}}{\omega_i^2 - \omega^2 + 2s\omega_i D_{L,i}}, \quad (4.30)$$

wobei $[\bar{\mathbf{C}}_p + s\bar{\mathbf{C}}_v]_{*i} \cdot [\bar{\mathbf{F}}]_{i*}$ das dyadische Produkt der i -ten Spalte der Matrix $(\bar{\mathbf{C}}_p + s\bar{\mathbf{C}}_v)$ und der i -ten Zeile von $\bar{\mathbf{F}}$ sowie ω_i und $D_{L,i}$ die i -te Eigenfrequenz bzw. das i -te Lehrsche Dämpfungsmaß bezeichnet.

Die Übertragungsfunktion erhält man als Summe der Anteile der einzelnen Eigenmoden

$$\mathbf{H}(s) = \sum_{i=1}^n \mathbf{H}_i(s) = \sum_{i=1}^n \frac{[\bar{\mathbf{C}}_p + s\bar{\mathbf{C}}_v]_{*i} \cdot [\bar{\mathbf{F}}]_{i*}}{\omega_i^2 - \omega^2 + 2s\omega_i D_{L,i}}. \quad (4.31)$$

Kapitel 5

Grundlagen der Modellreduktion

Dieses Kapitel stellt die Grundlagen für die Methoden und Konzepte der Verfahren zur Modellreduktion vor. In Abschnitt 5.1 erfolgt die Problemdefinition der Modellreduktion, wobei auch ausführlich auf die Ansprüche an ein reduziertes Modell eingegangen wird. Da in dieser Arbeit Reduktionsverfahren, die auf Projektion basieren betrachtet werden, erfolgt im darauf folgenden Abschnitt 5.2 die Beschreibung der Modellreduktion durch Projektion.

5.1 Problemstellung der Modellreduktion

In diesem Abschnitt wird die Problemstellung der Modellreduktion beschrieben. Dazu wird zunächst das Problem der Modellreduktion näher erläutert. In einem weiteren Abschnitt werden die grundsätzlichen Ansprüche, die an ein Reduktionsverfahren gestellt werden, dargestellt.

5.1.1 Problemstellung

Eine Modellreduktion ist überall dort von Nöten, wo sehr große Modelle zur Anwendung kommen. Große Modelle entstehen in vielen Anwendungsgebieten wie z.B. beim Entwurf von integrierten Schaltungen, in der Mikrosystemtechnik, der Luft und Raumfahrt, der Erdbebenforschung oder bei der Untersuchung von Schwingungsproblemen. Die Modelle dienen dabei der Analyse, dem Entwurf, der Optimierung von Systemen oder dem Reglerentwurf.

Sollen oder müssen bestimmte Eigenschaften einer komplexen physikalischen Struktur durch ein Modell mit einem hohen Detaillierungsgrad abgebildet werden, resultiert dies üblicherweise in Differenzialgleichungssysteme, die sehr große Dimensionen aufweisen. Trotz gesteigerter Rechenleistung der Computertechnik ist eine Simulation mit diesen Systemen meist nicht praktikabel. Vor einer weiteren Verwendung der Modelle ist somit eine Reduktion der Dimension der zugrunde liegenden Systeme meist unumgänglich.

Ein Anwendungsgebiet, in dem die Reduzierung von Modellen unumgänglich ist, ist die Simulation von Werkzeugmaschinen. Wie in Abbildung 5.1 dargestellt, wird die Reduktion nach Aufstellen eines hochdimensionalen Modells des mechanischen Systems durchgeführt.

Bei der Modellreduktion eines Systems wird ein System mit kleinerer Dimension gesucht, das eine

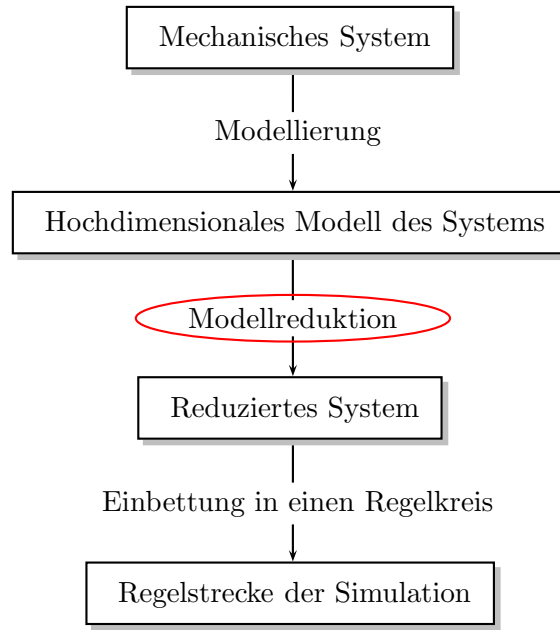


Abbildung 5.1: Einsatz der Modellreduktion bei der Simulation von Werkzeugmaschinen

Berechnung einer Lösung des Systems mit geringerem Zeit- und Speicheraufwand ermöglicht und dabei die Eigenschaften des Originalsystems möglichst gut approximiert. Die Eigenschaften des Originalsystems wie Stabilität, Steuerbarkeit und Beobachtbarkeit sollen dabei erhalten bleiben. Das Ziel des Reduktionsprozesses besteht darin, ein reduziertes System $\hat{\Sigma}$ mit Ausgang $\hat{\mathbf{y}}(t)$ zu ermitteln, so dass $\|\mathbf{y}(t) - \hat{\mathbf{y}}(t)\|$ für alle zulässigen Eingangssignale $\mathbf{u}(t)$, bezüglich einer geeignet gewählten Norm $\|\cdot\|$, klein ist. Für die in dieser Arbeit betrachteten Systeme bedeutet dies die Reduzierung der originalen Systemmatrizen auf eine Dimension $r < 100$ bei möglichst guter Approximation der Übertragungsfunktion im unteren Frequenzband bis 500 Hz. Weiterhin soll die Impulsantwort bis zu einer Zeit von 0,5 Sekunden gut approximiert werden.

Neben dieser Anforderung soll ein reduziertes System ggf. weitere spezielle Anforderungen erfüllen, die durch das Anwendungsgebiet, für die die Reduktion durchgeführt wird, bestimmt werden. Die speziellen Anforderungen an reduzierte Systeme im Bereich der Simulation von Werkzeugmaschinen, werden in Abschnitt 10.2 näher dargestellt.

Mit diesen Vorgaben lässt sich das Problem der Modellreduktion wie folgt formulieren:

Problem 5.1 Gegeben sei das lineare zeitinvariante (LTI) System in der Zustandsraumdarstellung:

$$\Sigma_{1o} := \begin{cases} \mathbf{E}\dot{\mathbf{z}}(t) = \mathbf{A}\mathbf{z}(t) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) = \mathbf{C}\mathbf{z}(t), \end{cases} \quad (5.1)$$

mit $\mathbf{E}, \mathbf{A} \in \mathbb{R}^{2n \times 2n}$, $\mathbf{B} \in \mathbb{R}^{2n \times m}$ und $\mathbf{C} \in \mathbb{R}^{q \times 2n}$. Bei der Reduktion eines Systems erster Ordnung der Form (5.1) wird ein System mit $r \ll 2n$ gesucht, das folgende Form aufweist:

$$\hat{\Sigma}_{1o} := \begin{cases} \hat{\mathbf{E}}\dot{\hat{\mathbf{z}}}(t) = \hat{\mathbf{A}}\hat{\mathbf{z}}(t) + \hat{\mathbf{B}}\mathbf{u}(t), \\ \hat{\mathbf{y}}(t) = \hat{\mathbf{C}}\hat{\mathbf{z}}(t), \end{cases} \quad (5.2)$$

wobei $\hat{\mathbf{E}}, \hat{\mathbf{A}} \in \mathbb{R}^{r \times r}$, $\hat{\mathbf{B}} \in \mathbb{R}^{r \times m}$ und $\hat{\mathbf{C}} \in \mathbb{R}^{q \times r}$. Die reduzierten Systemmatrizen und der Zustand $\hat{\mathbf{z}}(t)$ hat im reduzierten System die Dimension r . Der Eingang $\mathbf{u}(t)$ bleibt unverändert und der Ausgang $\hat{\mathbf{y}}(t)$ stellt eine Approximation an den Ausgang $\mathbf{y}(t)$ des Originalsystems dar.

Neben der Reduktion des Systems in Form eines Systems erster Ordnung ist auch die Reduktion des Systems in Form eines Systems zweiter Ordnung möglich.

Problem 5.2 Gegeben sei das lineare zeitinvariante (LTI) dynamische System in der Darstellung als System zweiter Ordnung:

$$\Sigma_{2o} := \begin{cases} \mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{D}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) = \mathbf{F}\mathbf{u}(t), \\ \mathbf{y}(t) = \mathbf{C}_p\mathbf{x}(t) + \mathbf{C}_v\dot{\mathbf{x}}(t) \end{cases} \quad (5.3)$$

mit $\mathbf{M}, \mathbf{D}, \mathbf{K} \in \mathbb{R}^{n \times n}$, $\mathbf{F} \in \mathbb{R}^{n \times m}$ und $\mathbf{C}_p, \mathbf{C}_v \in \mathbb{R}^{q \times n}$. Bei der Reduktion eines Systems zweiter Ordnung der Form (5.3) wird ein System mit $r \ll n$ gesucht, das folgende Form aufweist:

$$\hat{\Sigma}_{2o} := \begin{cases} \hat{\mathbf{M}}\ddot{\hat{\mathbf{x}}}(t) + \hat{\mathbf{D}}\dot{\hat{\mathbf{x}}}(t) + \hat{\mathbf{K}}\hat{\mathbf{x}}(t) = \hat{\mathbf{F}}\mathbf{u}(t), \\ \hat{\mathbf{y}}(t) = \hat{\mathbf{C}}_p\hat{\mathbf{x}}(t) + \hat{\mathbf{C}}_v\dot{\hat{\mathbf{x}}}(t) \end{cases} \quad (5.4)$$

wobei $\hat{\mathbf{M}}, \hat{\mathbf{D}}, \hat{\mathbf{K}} \in \mathbb{R}^{r \times r}$, $\hat{\mathbf{F}} \in \mathbb{R}^{r \times m}$ und $\hat{\mathbf{C}}_p, \hat{\mathbf{C}}_v \in \mathbb{R}^{q \times r}$. Die reduzierten Systemmatrizen $\hat{\mathbf{M}}, \hat{\mathbf{D}}, \hat{\mathbf{K}}$ und die Zustände $\hat{\mathbf{x}}(t)$ haben im reduzierten System die Dimension $r \ll n$. Der Eingang $\mathbf{u}(t)$ bleibt unverändert und der Ausgang $\hat{\mathbf{y}}(t)$ stellt eine Approximation an den Ausgang $\mathbf{y}(t)$ des Originalsystems dar.

Der Vorteil der Reduktion eines Systems zweiter Ordnung und nachfolgender Transformation in ein System erster Ordnung liegt darin, dass die Dimension des Systems zweiter Ordnung nur halb so groß ist wie die Dimension des Systems erster Ordnung, wodurch die Berechnung eines reduzierten Systems und die Transformation in ein System erster Ordnung weniger Speicherplatz in Anspruch nehmen. Zudem ist es möglich, das System erst ohne Dämpfung zu reduzieren und anschließend mit einer proportionalen Dämpfung als Linearkombination (4.23) der reduzierten Massen- und Steifigkeitsmatrix oder mit einer modalen Dämpfung zu versehen. Bei einer Änderung der Dämpfungsparameter ist somit keine erneute Reduktion des Systems notwendig.

5.1.2 Anspruch an das Reduktionsverfahren

Soll ein Reduktionsverfahren auf die hier betrachteten Strukturmodelle angewendet werden, die zur Simulation des Verhaltens von Werkzeugmaschinen dienen, so muss dieses einige besondere Bedingungen erfüllen. Um detaillierte Aussagen über das Verhalten der Strukturmodelle machen zu können, sind Vernetzungen mit einer hohen Anzahl an Gitterpunkten notwendig. Dies führt zu Modellen mit einer hohen Anzahl an Freiheitsgraden. Dies spiegelt sich wiederum in einer hohen Anzahl an Systemgleichungen wider. Die zu reduzierenden Systeme besitzen deshalb hochdimensionale aber dünnbesetzte Systemmatrizen. Um ein solches System effizient reduzieren zu können, muss das Reduktionsverfahren die besondere Struktur der Systemmatrizen ausnutzen.

Neben einem automatischen Reduktionsprozeß werden weitere Anforderungen an das Reduktionsverfahren gestellt, so sollen z. B. wichtige Struktureigenschaften, wie die Stabilität des Originalsystems, auf das reduzierte System übertragen werden.

Durch die Verwendung kommerzielle Simulationsumgebungen zur Simulation von Werkzeugmaschinen werden an das reduzierte System spezielle Bedingungen geknüpft. Zur Simulation von Werkzeugmaschinen werden üblicherweise kommerzielle Simulationsumgebungen verwendet, die ausschließlich mit reellen Systemmatrizen operieren. Deshalb muss das Reduktionsverfahren reelle Projektionsmatrizen generieren können, um mit deren Hilfe ein reelles reduziertes System zu erhalten.

In der Theorie führt die in dieser Arbeit beschriebene FEM-Modellierung der Teilkomponenten einer Werkzeugmaschine zu Modellen, die durch ein stabiles System beschrieben werden. Durch numerische Rundungsfehler kann jedoch nicht ausgeschlossen werden, dass in der Praxis Modelle generiert werden, die durch instabile Systeme beschrieben werden.

In solchen Fällen ist es möglich, das Originalsystem in ein instabiles reduziertes System zu reduzieren und anschließend mit Methoden wie z.B. [81, 101], die instabilen Subsysteme aus dem reduzierten System eliminieren, in ein stabiles System zu überführen. Allerdings ist ein Reduktionsverfahren, das instabile Systeme zu stabilen reduzierten Systemen reduziert vom Vorteil.

5.2 Modellreduktion durch Projektion

Die Verfahren, die in dieser Arbeit verwendet werden, basieren auf einer Reduktion durch Projektion. Die Vorgehensweise um ein reduziertes Modell durch ein Verfahren, das auf Projektion basiert, zu berechnen, wird hier zunächst am Beispiel eines Systems erster Ordnung der Form (3.4) dargestellt. Dabei orientiert sich die Darstellung der Vorgehensweise an [111].

Dazu seien $\mathcal{V}, \mathcal{W} \subseteq \mathbb{K}^{2n}$ zwei gegebene Unterräume der Dimension r . Im ersten Schritt wird die Lösung des Systems $\mathbf{z}(t) \in \mathbb{K}^{2n}$ im Unterraum \mathcal{V} der Dimension r approximiert. Dieser Unterraum wird durch die Matrix $\mathbf{V}_r \in \mathbb{K}^{2n \times r}$ mit $\text{colspan}(\mathbf{V}_r) = \mathcal{V}$ dargestellt, wobei gilt

$$\mathbf{z}(t) \approx \mathbf{V}_r \hat{\mathbf{z}}(t). \quad (5.5)$$

Das Einsetzen von (5.5) in das System (3.4) führt auf das überbestimmte System

$$\begin{aligned} \mathbf{E}\mathbf{V}_r \dot{\hat{\mathbf{z}}}(t) &= \mathbf{A}\mathbf{V}_r \hat{\mathbf{z}}(t) + \mathbf{B}\mathbf{u}(t) + \boldsymbol{\epsilon}(t), \\ \mathbf{y}(t) &\approx \mathbf{C}\mathbf{V}_r \hat{\mathbf{z}}(t). \end{aligned} \quad (5.6)$$

Ist $\mathbf{z}(t) = \mathbf{V}_r \hat{\mathbf{z}}(t)$ die exakte Lösung des Systems (3.4), dann ist $\boldsymbol{\epsilon}(t) = 0$ für alle t . Da aber im Allgemeinen die exakte Lösung $\mathbf{z}(t)$ kein Element des Unterraumes \mathcal{V} ist, bleibt ein Residuum $\boldsymbol{\epsilon} \neq 0$ bestehen. Um eine eindeutige Lösung des Systems (5.6) zu erhalten, werden r Nebenbedingungen an das Residuum gestellt. Diese werden üblicherweise als Orthogonalitätsbedingungen des Residuums bezüglich des zweiten Unterraumes \mathcal{W} formuliert und *Petrov-Galerkin-Bedingung* genannt:

$$\mathbf{E}\mathbf{V}_r \dot{\hat{\mathbf{z}}}(t) - \mathbf{A}\mathbf{V}_r \hat{\mathbf{z}}(t) - \mathbf{B}\mathbf{u}(t) \perp \mathcal{W}, \quad \text{für alle } t. \quad (5.7)$$

Sei \mathbf{W}_r eine Matrix, dessen Spalten eine Basis des Unterraumes \mathcal{W} aufspannen, d.h. $\text{colspan}(\mathbf{W}_r) = \mathcal{W}$, dann kann die *Petrov-Galerkin-Bedingung* in der Form

$$\mathbf{W}_r^*(\mathbf{E}\mathbf{V}_r \dot{\hat{\mathbf{z}}}(t) - \mathbf{A}\mathbf{V}_r \hat{\mathbf{z}}(t) - \mathbf{B}\mathbf{u}(t)) = 0, \quad \text{für alle } t \quad (5.8)$$

geschrieben werden. Unter der Annahme, dass $\mathbf{W}_r^* \mathbf{V}_r$ nicht singulär ist, erhält man nach einer Multiplikation der Gleichung (5.8) von links mit $(\mathbf{W}_r^* \mathbf{V}_r)^{-1}$:

$$(\mathbf{W}_r^* \mathbf{V}_r)^{-1} \mathbf{W}_r^* \mathbf{E} \mathbf{V}_r \dot{\hat{\mathbf{z}}}(t) - (\mathbf{W}_r^* \mathbf{V}_r)^{-1} \mathbf{W}_r^* \mathbf{A} \mathbf{V}_r \hat{\mathbf{z}}(t) - (\mathbf{W}_r^* \mathbf{V}_r)^{-1} \mathbf{W}_r^* \mathbf{B} \mathbf{u}(t) = 0, \quad (5.9)$$

für alle t .

Dies liefert das reduzierte System erster Ordnung (5.2) mit den Systemmatrizen

$$\begin{aligned} \hat{\mathbf{E}} &= \mathbf{W}^* \mathbf{E} \mathbf{V}, & \hat{\mathbf{A}} &= \mathbf{W}^* \mathbf{A} \mathbf{V}, \\ \hat{\mathbf{B}} &= \mathbf{W}^* \mathbf{B}, & \hat{\mathbf{C}} &= \mathbf{C} \mathbf{V} \end{aligned} \quad (5.10)$$

mit

$$\mathbf{V} = \mathbf{V}_r \quad \text{und} \quad \mathbf{W}^* = (\mathbf{W}_r^* \mathbf{V}_r)^{-1} \mathbf{W}_r^*. \quad (5.11)$$

Sind die beiden Matrizen \mathbf{V}_r und \mathbf{W}_r biorthogonal, d.h. gilt $\mathbf{W}_r^* \mathbf{V}_r = \mathbf{I}$, dann vereinfachen sich die Gleichungen (5.11) zu:

$$\mathbf{V} = \mathbf{V}_r \quad \text{und} \quad \mathbf{W}^* = \mathbf{W}_r^*. \quad (5.12)$$

In Abbildung 5.2 ist die Reduktion durch Projektion eines Systems erster Ordnung grafisch dargestellt.

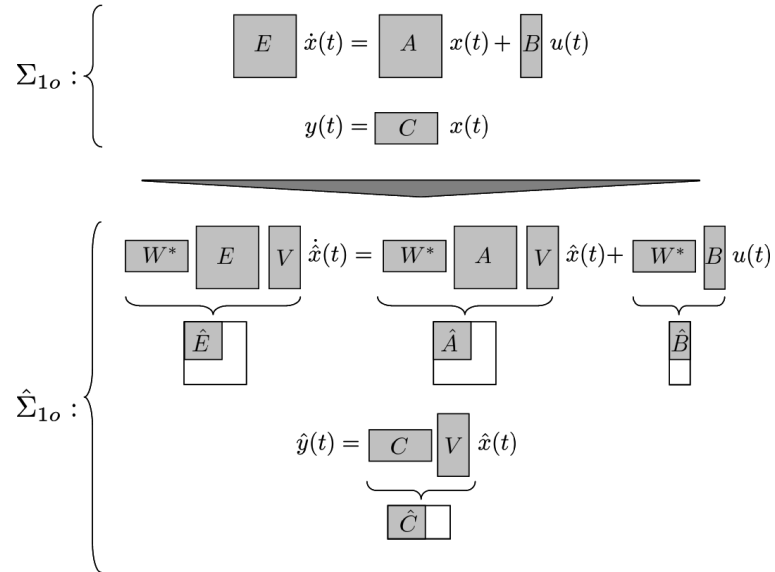


Abbildung 5.2: Reduktion eines Systems erster Ordnung durch Projektion

Der folgende Satz zeigt, dass die Wahl der Basen der Unterräume \mathcal{V} und \mathcal{W} für die Berechnung des reduzierten Systems nicht von Bedeutung ist.

Satz 5.1 (Äquivalenz reduzierter Systeme) *Bilden die Spaltenvektoren der Matrizen $\tilde{\mathbf{V}}$ und $\tilde{\mathbf{W}}$ ebenfalls eine Basis der Vektorräume \mathcal{V} bzw. \mathcal{W} , dann ist das mit diesen Matrizen reduzierte System äquivalent mit dem durch die Matrizen \mathbf{V} und \mathbf{W} reduzierten System (5.10).*

BEWEIS: Siehe z. B. [81, 121]. □

Werden in den Gleichungen (5.10) zwei unterschiedliche Matrizen \mathbf{V}_r und \mathbf{W}_r zu den entsprechenden Unterräumen \mathcal{V} bzw. \mathcal{W} verwendet, handelt es sich um eine *Petrov-Galerkin-Projektion*. Wird hingegen $\mathbf{W}_r = \mathbf{V}_r$ mit $\mathbf{V}_r^* \mathbf{V}_r = \mathbf{I}$ verwendet, handelt es sich um eine *Galerkin-Projektion*, siehe Abschnitt 2.5. Die anhand von Systemen erster Ordnung beschriebene Vorgehensweise kann auf Systeme zweiter Ordnung übertragen werden. Dazu seien $\mathcal{V}, \mathcal{W} \subseteq \mathbb{K}^n$ zwei gegebene Unterräume der Dimension r . Betrachtet man ein System zweiter Ordnung der Form (3.1) und setzt für den Lösungsvektor $\mathbf{x}(t) \in \mathbb{K}^n$

$$\mathbf{x}(t) \approx \mathbf{V}_r \hat{\mathbf{x}}(t), \quad (5.13)$$

wobei $\mathbf{V}_r \in \mathbb{K}^{n \times r}$, dann erhält man durch Einsetzen von (5.13) in das System (3.1) das überbestimmte System

$$\begin{aligned} \mathbf{M}\mathbf{V}_r \ddot{\hat{\mathbf{x}}}(t) + \mathbf{D}\mathbf{V}_r \dot{\hat{\mathbf{x}}}(t) + \mathbf{K}\mathbf{V}_r \hat{\mathbf{x}}(t) &= \mathbf{F}\mathbf{u}(t) + \boldsymbol{\epsilon}(t), \\ \mathbf{y}(t) &\approx \mathbf{C}_p \mathbf{V}_r \hat{\mathbf{x}}(t) + \mathbf{C}_v \mathbf{V}_r \dot{\hat{\mathbf{x}}}(t). \end{aligned} \quad (5.14)$$

Dies führt mit einer Matrix $\mathbf{W}_r \in \mathbb{K}^n$, so dass $\mathbf{W}_r^* \mathbf{V}_r$ nicht singulär ist, zu der *Petrov-Galerkin-Bedingung*

$$\begin{aligned} (\mathbf{W}_r^* \mathbf{V}_r)^{-1} \mathbf{W}_r^* \mathbf{M}\mathbf{V}_r \ddot{\hat{\mathbf{z}}}(t) + (\mathbf{W}_r^* \mathbf{V}_r)^{-1} \mathbf{W}_r^* \mathbf{D}\mathbf{V}_r \dot{\hat{\mathbf{z}}}(t) + (\mathbf{W}_r^* \mathbf{V}_r)^{-1} \mathbf{W}_r^* \mathbf{K}\mathbf{V}_r \hat{\mathbf{z}}(t) \\ - (\mathbf{W}_r^* \mathbf{V}_r)^{-1} \mathbf{W}_r^* \mathbf{F}\mathbf{u}(t) = 0, \end{aligned} \quad (5.15)$$

für alle t .

Dies liefert das reduzierte System zweiter Ordnung (5.4) mit den Systemmatrizen

$$\begin{aligned} \hat{\mathbf{M}} &= \mathbf{W}^* \mathbf{M}\mathbf{V}, \quad \hat{\mathbf{D}} = \mathbf{W}^* \mathbf{D}\mathbf{V}, \quad \hat{\mathbf{K}} = \mathbf{W}^* \mathbf{K}\mathbf{V}, \\ \hat{\mathbf{F}} &= \mathbf{W}^* \mathbf{F}, \quad \hat{\mathbf{C}}_p = \mathbf{W}^* \mathbf{C}_p \mathbf{V}, \quad \hat{\mathbf{C}}_v = \mathbf{W}^* \mathbf{C}_v \mathbf{V} \end{aligned} \quad (5.16)$$

mit

$$\mathbf{V} = \mathbf{V}_r \quad \text{und} \quad \mathbf{W}^* = (\mathbf{W}_r^* \mathbf{V}_r)^{-1} \mathbf{W}_r^*. \quad (5.17)$$

Sind die beiden Matrizen \mathbf{V}_r und \mathbf{W}_r biorthogonal, d.h. gilt $\mathbf{W}_r^* \mathbf{V}_r = \mathbf{I}$, dann vereinfachen sich die Gleichungen (5.17) zu:

$$\mathbf{W}^* = \mathbf{W}_r^* \quad \text{und} \quad \mathbf{V} = \mathbf{V}_r. \quad (5.18)$$

In Abbildung 5.3 ist die Reduktion durch Projektion eines Systems zweiter Ordnung grafisch dargestellt.

Wird ein Projektionsverfahren auf ein System erster Ordnung in der Form einer verallgemeinerten Zustandsdarstellung der Form (3.4) angewendet, das aus der Transformation eines Systems zweiter Ordnung der Form (3.1) entstanden ist, dann kann im Allgemeinen nicht davon ausgegangen werden, dass die spezielle Struktur der Systemmatrizen \mathbf{E} und \mathbf{A} der Zustandsdarstellung, im reduzierten System (5.10) erhalten bleibt. Eine hinreichende Bedingung dafür ist die Verwendung blockdiagonaler Projektionsmatrizen

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_r & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_r \end{bmatrix} \in \mathbb{K}^{2n \times 2r}, \quad \mathbf{W} = \begin{bmatrix} \mathbf{W}_r & \mathbf{0} \\ \mathbf{0} & \mathbf{W}_r \end{bmatrix} \in \mathbb{K}^{2n \times 2r}, \quad (5.19)$$

$$\begin{array}{c}
 \Sigma_{2o} : \left\{ \begin{array}{l} \boxed{M} \ddot{x}(t) + \boxed{D} \dot{x}(t) + \boxed{K} x(t) = \boxed{F} u(t) \\ y(t) = \boxed{C_p} x(t) + \boxed{C_v} \dot{x}(t) \end{array} \right. \\
 \hline
 \hat{\Sigma}_{2o} : \left\{ \begin{array}{l} \underbrace{\boxed{W^*} \boxed{M} \boxed{V}}_{\boxed{\hat{M}}} \ddot{\hat{x}}(t) + \underbrace{\boxed{W^*} \boxed{D} \boxed{V}}_{\boxed{\hat{D}}} \dot{\hat{x}}(t) + \underbrace{\boxed{W^*} \boxed{K} \boxed{V}}_{\boxed{\hat{K}}} \hat{x}(t) = \underbrace{\boxed{W^*} \boxed{F}}_{\boxed{\hat{F}}} u(t) \\ \hat{y}(t) = \underbrace{\boxed{C_p} \boxed{V}}_{\boxed{\hat{C}_p}} \hat{x}(t) + \underbrace{\boxed{C_v} \boxed{V}}_{\boxed{\hat{C}_v}} \dot{\hat{x}}(t) \end{array} \right.
 \end{array}$$

Abbildung 5.3: Reduktion eines Systems zweiter Ordnung durch Projektion

mit $\mathbf{V}_r, \mathbf{W}_r \in \mathbb{K}^{n \times r}$ in (5.11).

Ist darüber hinaus das Produkt $\mathbf{W}_r^* \mathbf{V}_r = \mathbf{I}$ regulär und liegt nach der Transformation des Systems zweiter Ordnung der Form (3.1) ein System erster Ordnung in allgemeiner Form mit den Systemmatrizen (3.5) und $\mathbf{N} = \mathbf{I}$ vor, so kann das erhaltene reduzierte System wieder direkt als System zweiter Ordnung mit den reduzierten Systemmatrizen

$$\begin{aligned}
 \hat{\mathbf{M}} &= \mathbf{W}_r^* \mathbf{M} \mathbf{V}_r, & \hat{\mathbf{D}} &= \mathbf{W}_r^* \mathbf{D} \mathbf{V}_r, & \hat{\mathbf{K}} &= \mathbf{W}_r^* \mathbf{K} \mathbf{V}_r, \\
 \hat{\mathbf{F}} &= \mathbf{W}_r^* \mathbf{F}, & \hat{\mathbf{C}}_p &= \mathbf{C}_p \mathbf{V}_r, & \hat{\mathbf{C}}_v &= \mathbf{C}_v \mathbf{V}_r.
 \end{aligned} \tag{5.20}$$

geschrieben werden.

Teil II

Werkzeugmaschinen

Kapitel 6

Aufbau und Funktionsweise von Werkzeugmaschinen

Im Werkzeugmaschinenbau haben sich mechatronische Systeme den rein mechanischen Konstruktionen zunehmend als überlegen erweisen. In modernen Werkzeugmaschinen werden deshalb bereits neben den Regeleinrichtungen auch die Komponenten zur Ansteuerung der Leistungselektronik in zunehmenden Maße digital realisiert. Diese Entwicklung wurde und wird weiterhin durch die Leistungszunahme von Mikroprozessoren und den Fortschritten in den Bereichen der Informations- und Elektrotechnik vorangetrieben. Aus diesem Grund stellen moderne Werkzeugmaschinen in ihrer heutigen Ausprägung Beispiele für mechatronische Systeme dar.

In diesem Kapitel wird die Ausprägung der in dieser Arbeit betrachteten Werkzeugmaschinen dargestellt, wobei die Beschreibung des Aufbaus und der Funktionsweise moderner Werkzeugmaschinen im Mittelpunkt steht. Die zur Unterstützung der Auslegung von Werkzeugmaschinen nötige Simulation wird im folgenden Kapitel näher dargestellt.

Die Themengebiete der beiden folgenden Kapitel stehen nicht im Fokus dieser Arbeit, weshalb an dieser Stelle der Arbeit nur auf die Grundlagen des Aufbaus und der Funktionsweise von Werkzeugmaschinen eingegangen werden kann, deren Kenntnis zum Verständnis der nachfolgenden Kapitel beitragen sollen. Zu diesem Zweck werden in den beiden folgenden Kapitel Beschreibungen und Darstellungen verwendet, die den Beschreibungen und Darstellungen des Aufbaus und der Funktionsweise von Werkzeugmaschinen in [138] entlehnt sind.

6.1 Aufbau und Funktionsweise

Die Grundlage einer Werkzeugmaschine bildet immer ein mechanisches Grundsystem, das sogenannte Maschinengestell. Dabei handelt es sich um einen kinematischen Bewegungsapparat der spezifische Freiheitsgrade aufweist. Als Beispiel für ein solches Grundgestell ist in Abbildung 6.1 das Maschinengestell eines Bearbeitungszentrum mit fünf Vorschubachsen dargestellt [138]. Dabei handelt es sich um eine Maschine mit serieller Achskinematik, d.h. die Vorschubachsen (X,Y,Z,B,C) bilden eine kinematische Kette. Bei der Bewegung einer Vorschubachse werden die in der Kette nachgelagerten Schlitten als auch das Werkzeug bzw. das Werkstück mitgeführt.

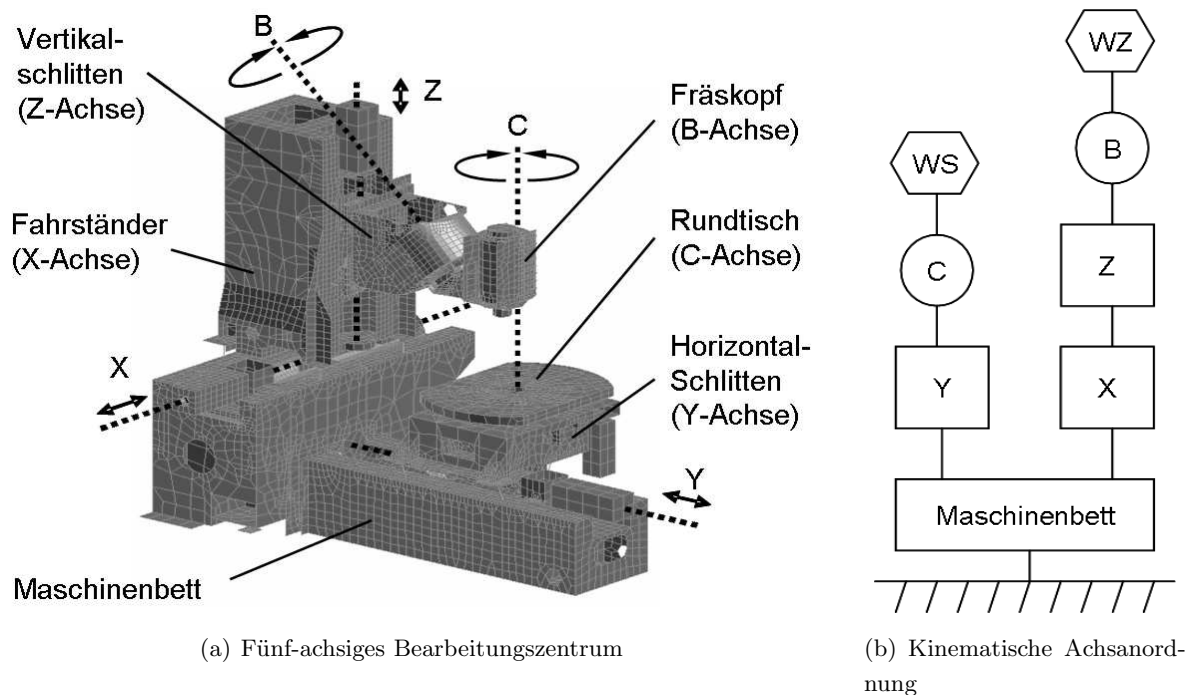


Abbildung 6.1: Werkzeugmaschine mit serieller Achskinematik nach [138]

Neben den seriellen Achsanordnungen sind die parallel geführten Kinematiken zu benennen. Bei diesen Maschinen wirken mehrere Stellantriebe auf das bewegte Maschinenelement (z.B. Werkzeugaufnahme, Maschinentisch) ein. Die Koordination der Antriebe zur Erzeugung gerichteter Bewegungen erfordert spezielle Algorithmen, wodurch erweiterte Anforderungen an die Steuerungstechnik gestellt werden. Parallelkinetische Antriebskonzepte führen damit zu anders gearteten Bewegungsapparaten, deren Entwurf und Analyse nach eigenen Kriterien erfolgt. In dieser Arbeit werden deshalb nur Maschinen mit klassischer Achskinematik betrachtet.

In Abhängigkeit von der Kombination translatorischer und rotatorischer Vorschubachsen besitzen das Werkzeug bzw. das Werkstück spezifische Freiheitsgrade und können über die Vorschubantriebe auf räumlichen Bahnkurven relativ zueinander bewegt werden. Die Trajektorien des Werkzeugs relativ zum Werkstück entstehen dabei durch die Überlagerung von interpolierten, geregelten Achsbewegungen, welche unter den Vorgaben der numerischen Steuerung (CNC¹) durch die positionsgeregelten elektrischen Vorschubantriebe erzeugt werden.

Die Werkzeugposition einer Werkzeugmaschine wird über den sogenannten *Tool Center Point* (TCP) beschrieben. Dies ist ein Punkt, der sich an geeigneter Stelle am Werkzeug befindet und auf den sich Berechnungen oder Messungen beziehen. Um zu beschreiben, welche Lage das Werkzeug einnehmen soll, genügt es, die Position und Orientierung des TCP im Raum zu definieren.

Die Abbildung 6.2 stellt das technologische Wirkprinzip von CNC-Maschinen anhand eines dreiachsigen Systems dar. Wie dargestellt lässt sich das Wirkprinzip von CNC-gesteuerten Werkzeugmaschinen in die fünf Ebenen der Prozesstechnik, der Mechanik, der Elektrotechnik, der

¹Englisch: Computerized Numerical Control

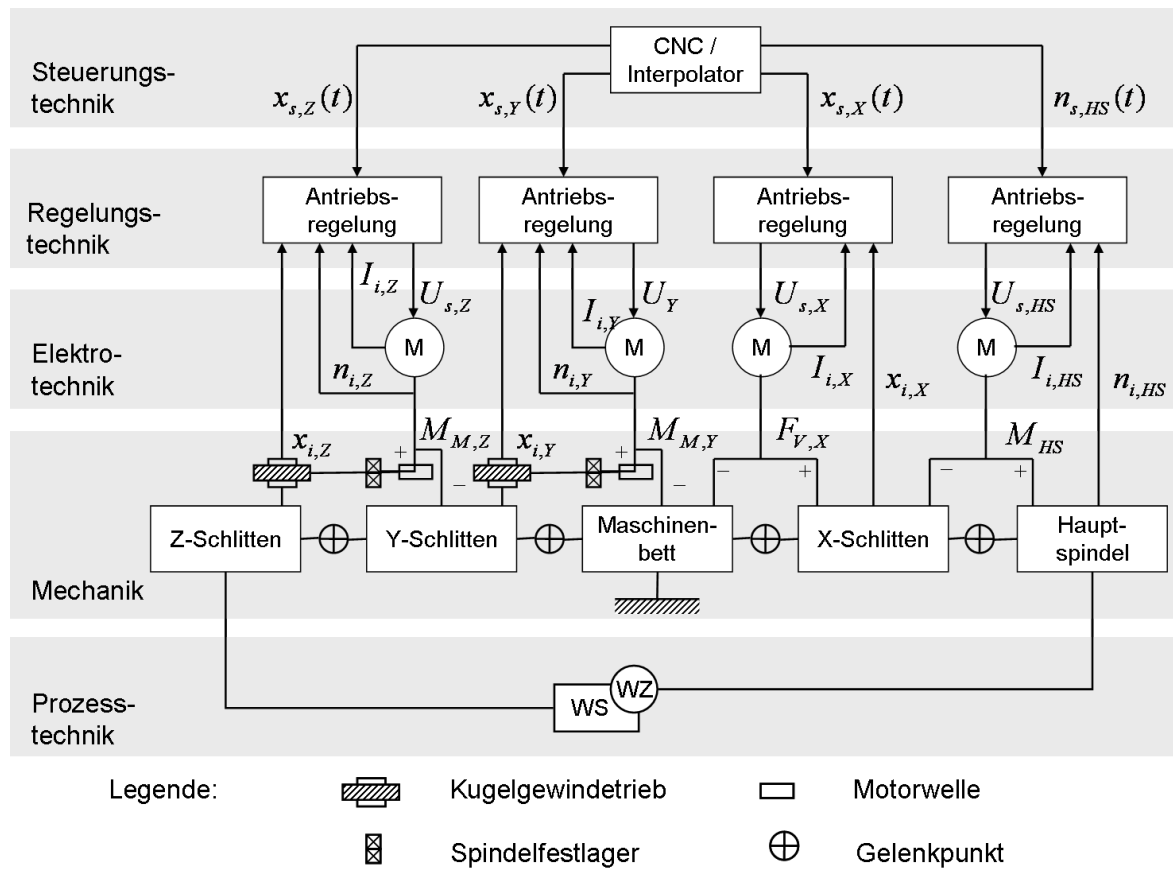


Abbildung 6.2: Wirkprinzip von CNC-gesteuerten Werkzeugmaschinen mit serieller Kinematik und lagegeregelten elektrischen Antrieben nach [138]

Regelungstechnik und der Steuerungstechnik gliedern. Auf der Ebene der Mechanik ist die kinematische Kette mit Z- und Y-Schlitten, Maschinenbett und X-Schlitten sowie die meist drehzahlgeführte ($n_{s,HS}$) Hauptspindel (HS) zu sehen. Die Hauptspindel führt das Werkzeug und der Schlitten am gegenüber liegenden Ende der kinematischen Kette das Werkstück. Ein in der Steuerung integrierter Interpolator berechnet anhand der programmierten Bewegungsanweisungen die Lagesollwerte $x_{s,k}$ für die einzelnen Achsantriebe ($k = X, Y, Z$). Für die bahngetreue Bewegung der Vorschubachsen auf den gewünschten Bahnkurven werden Antriebsregelungen mit geeigneten Algorithmen, die auf Messungen und Soll-Ist-Vergleichen der Schlittenposition $x_{i,k}$, Motordrehzahlen $n_{i,k}$ und Motorströme $I_{i,k}$ basieren, verwendet. Die erforderlichen Motormomente $M_{M,k}$ bzw. Vorschubkräfte $F_{V,k}$ werden von Servomotoren (M) erzeugt, die die dazu benötigte elektrische Leistung durch spannungsgeführte ($U_{s,k}$) Leistungsverstärker erhalten. Die linearen Vorschubbewegungen werden durch Kugelgewindtriebe (KGT) indirekt durch elektromechanische Antriebe, wie der Y- und Z-Schlitten in Abbildung 6.2, oder direkt durch Linearmotoren, wie der X-Schlitten in Abbildung 6.2, realisiert.

In den folgenden Abschnitten 6.2 und 6.3 wird auf den Aufbau und die Funktionsweise der elektrischen Vorschubantriebe bzw. der Antriebsregelung näher eingegangen. Zum Abschluss des Kapitels wird in Abschnitt 6.4 das dynamische Verhalten von Werkzeugmaschinen näher betrachtet.

6.2 Vorschubantriebe

6.2.1 Grundaufbau

Der Grundaufbau eines CNC-gesteuerten elektromechanischen Vorschubantriebes besteht in der CNC, der Achsregelmodule, der Leistungsbaugruppe und dem Antriebsbus. In Abbildung 6.3 ist der Aufbau eines CNC-gesteuerten elektromechanischen Vorschubantriebes mit Kugelgewindetrieb schematisch dargestellt. Der Antriebsbus dient der Kommunikation von Führungs-, Stell-

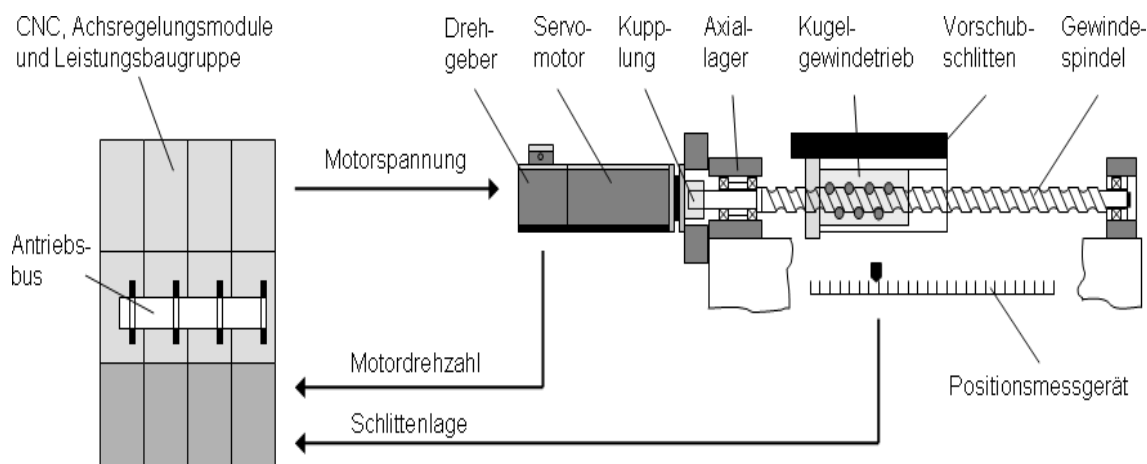


Abbildung 6.3: Aufbau eines CNC-gesteuerten elektromechanischen Vorschubantriebes (Kugelgewindetrieb) nach [138]

und Messgrößen. Die Leistungsbaugruppe besteht aus einem Ein- und Rückspeisemodul mit Gleichspannungszwischenkreis und Transistorstellern für die einzelnen Antriebsregelkreise. Diese Baugruppe versorgt die elektrischen Antriebe mit der benötigten elektrischen Energie und gehört mit dem Servomotor zu den elektrotechnischen Funktionselementen des Antriebssystems. Bei elektromechanischen Vorschubantrieben schließt sich dem Servomotor ein mechanisches Übertragungssystem an, dieses dient der Übertragung des vom Motor abgegebenen Momentes an den Vorschubschlitten, das durch ein mechanisches Prinzip in eine Vorschubkraft gewandelt wird.

6.2.2 Elektrische Antriebe

Als elektrische Antriebe werden heute vorwiegend permanenterregte Synchron-Drehstrommotoren eingesetzt [29]. Durch die Verwendung von Permanentmagneten am Rotor und der Ankerwicklungen im Stator entfällt beim Synchronmotor die mechanische Kommutierung. Dies führt zu einem Wegfall des mechanischen Verschleißes und zu einem verbesserten thermischen und dynamischen Verhalten. Dabei erfolgt die Kommutierung des permanenterregten Servomotor elektronisch. Durch die Verwendung moderner Regelungen, kann der Synchronmotor nach dem gleichen Prinzip wie der Gleichstrommotor angesteuert werden, wodurch er ein vergleichbares Regelverhalten aufweist [120, 159].

Das durch den Motor erzeugte Antriebsmoment wird durch das mechanische Übertragungs-

system übertragen und in eine Vorschubkraft gewandelt. Als Elemente zur Kraft-Momenten-Umsetzung kommen neben den typischen Übertragungselementen im Kraftfluss zwischen Motor und Schlitten wie Kupplungen, Getriebe, Lager und Führungen beispielsweise auch Kugelgewindetriebe oder Zahnstange-Ritzel-Systeme in Frage. Allgemeine Anforderungen, die im Hinblick auf ein gutes statisches und dynamisches Übertragungsverhalten an die Elemente von mechanischen Übertragungssystemen gestellt werden, sind eine hohe Steifigkeit, Spielfreiheit und geringe Trägheit [182].

6.2.3 Mechanische Übertragungssysteme

Bei Werkzeugmaschinen ist für die Realisierung von linearen Vorschubbewegungen der Kugelgewindetrieb am weitesten verbreitet, siehe Abbildung 6.3. Bei diesem Maschinenelement erfolgt die Umsetzung der Rotation in eine Linearbewegung durch die Gewindesteigung. Die in der Gewinderille umlaufenden Kugeln werden nach dem Austritt aus dem Eingriffsgebiet mittels eines Kugelumlenksystems durch die Mutter hindurch zurückgeführt, so dass sich in Bezug auf den Kugelumlauf ein unbegrenzter Fahrweg ergibt. Kugelgewindetriebe weisen einen hohen Wirkungsgrad (0,95-0,98) und eine hohe Lebensdauer auf [30].

Als Alternative zu Kugelgewindetrieben ist der Zahnstange-Ritzel-Antrieb zu nennen. Dieses System zeichnet sich dadurch aus, dass die Gesamtsteifigkeit des Systems unabhängig vom Arbeitsraum der Vorschubachse ist. Dieses System wird auf Grund dieser Eigenschaft bei Großwerkzeugen mit niedrigen Drehzahlen und hohen Drehmomenten eingesetzt [30].

Die heute verwendeten modernen Servomotoren sind leistungstark und über weite Drehzahlbereiche elektronisch steuerbar. Dies erlaubt häufig einen vollständigen Verzicht auf ein Getriebe zur Drehzahl- und Momentenanpassung. In diesem Fall wird die Kugelgewindespindel direkt über eine Kupplung angetrieben. Kann auf ein Getriebe nicht verzichtet werden, so sind heute neben Stirnradgetrieben auch Zahnriemengetriebe weit verbreitet, die sich durch gute Dämpfungs- und Geräuscheigenschaften auszeichnen.

Um die Momente und Kräfte übertragenden Wellen zu führen und die in radialer und axialer Richtung auftretenden Kräfte aufzunehmen dienen Lager. Dabei werden hinsichtlich der axialen Lagerung die Fest-Fest-, die Fest-Los- und die angestellte Lagerung als grundsätzliche Lageranordnungen unterschieden. Bei Kugellagern wird die höchste Gesamtsteifigkeit durch Fest-Fest-Lagerung und Vorspannung der Spindel mit Axial-Radial-Lagern erreicht.

6.2.4 Linear-Direktantriebe

Als Alternative zu den oben vorgestellten elektromechanischen Antriebssystemen hat sich in den letzten Jahren die Linear-Direktantriebstechnik erwiesen. Im Gegensatz zu den elektromechanischen Antriebssystemen mit rotatorischem Elektromotor und nachgeschaltetem mechanischem Übertragungssystem erfolgt bei diesen Antrieben die Krafterzeugung direkt am Vorschubschlitten, wodurch auf ein mechanisches Übertragungssystem verzichtet werden kann. Damit entfallen störende Elastizitäten bei der Kraftübertragung vom Motor auf den Schlitten, wodurch sich Vorteile hinsichtlich der Dynamik und der Regelung des Antriebes ergeben. Im Vergleich zu einem

elektromechanischem Antrieb ermöglicht ein Linear-Direktantrieb im Allgemeinen den Betrieb der Antriebsregelung mit deutlich höheren Proportionalverstärkungen, wodurch sich auch bei hohen Verfahrensgeschwindigkeiten höhere Positionier- und Bahngenauigkeiten realisieren lassen [138]. Die Abbildung 6.4 zeigt das Funktionsprinzip eines Synchron-Linearmotors. Dieser ent-

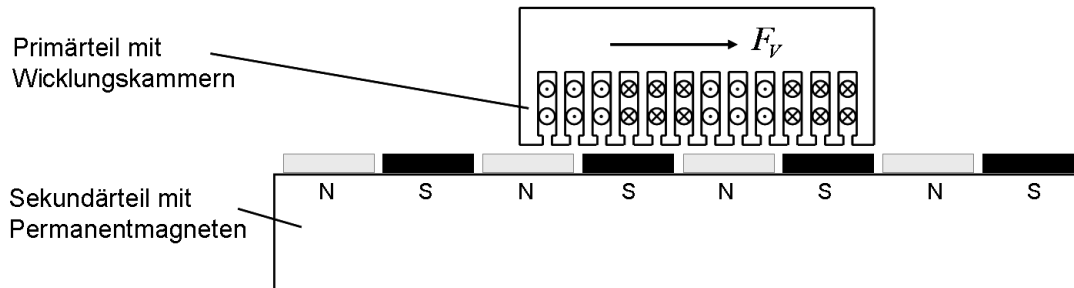


Abbildung 6.4: Funktionsprinzip eines Synchron-Linearmotors nach [147]

spricht einem um den Umfang abgewickelten Drehstrommotors. Dabei geht das Primärteil aus dem Stator der Drehstrommaschine hervor und beinhaltet die Phasenwicklungen. Neben den Vorteilen weist diese Antriebstechnik auch eine Reihe von Nachteile auf. So sind die geringen Wirkungsgrade mit einhergehenden thermischen Belastungen der Maschinenstruktur und der Motorwicklungen, die Linearführung belastenden hohen permanentmagnetischen Anziehungskräfte und die Begrenzung der Vorschubkräfte durch den Bauraum zu nennen. Aus den genannten Gründen sind die elektromechanischen Servoantriebssysteme mit Kugelgewindetrieb auch bei zunehmender Verbreitung von Linear-Direktantrieben nach wie vor die in Werkzeugmaschinen überwiegend eingesetzte Technik.

6.3 Antriebsregelung

Zur Positionsregelung der elektrischen Vorschubantriebe werden heute kaskadierte Regelkreisstrukturen verwendet [1, 160]. In Abbildung 6.5 ist eine solche kaskadierte Regelkreisstruktur dargestellt. Regelungen werden heute durchgehend digital realisiert. Dabei werden die Algorithmen der Regelung in Software implementiert und auf einer Recheneinheit eines Mikroprozessors des digitalen Regelungssystems abgearbeitet [97, 98]. Eine kaskadierte Regelkreisstruktur besteht dabei aus mehreren kaskadierten Regelkreisen, die auch Kaskadenebenen genannt werden. Bei der hier betrachteten Regelkreisstruktur handelt es sich um einen Lageregler, der auf der obersten Kaskadenebene angeordnet ist und durch zwei weitere untergelagerte Hilfsregelkreise zur Drehzahl- und Stromregelung unterstützt wird. Die schnelle und genaue Erfassung der Schlittenpositionen und Motorwinkellagen wird heute durch hochauflösende, optische Messgeräte erreicht. Ein Drehgeber an der Motorwelle erfasst dazu die Winkellage φ_i des Motors, die anhand der Übersetzungsverhältnisse im Antriebsstrang in eine Schlittenposition x_i umgerechnet werden kann. Höhere Positioniergenauigkeiten werden mit direkt am Vorschubschlitten montierten Längenmessgeräten erreicht.

Der Regelkreis der obersten Ebene mit der Lageregelung berechnet anhand der Differenz zwischen den Sollwerten x_s der Positionsvorgaben, die durch den Interpolator vorgegeben werden

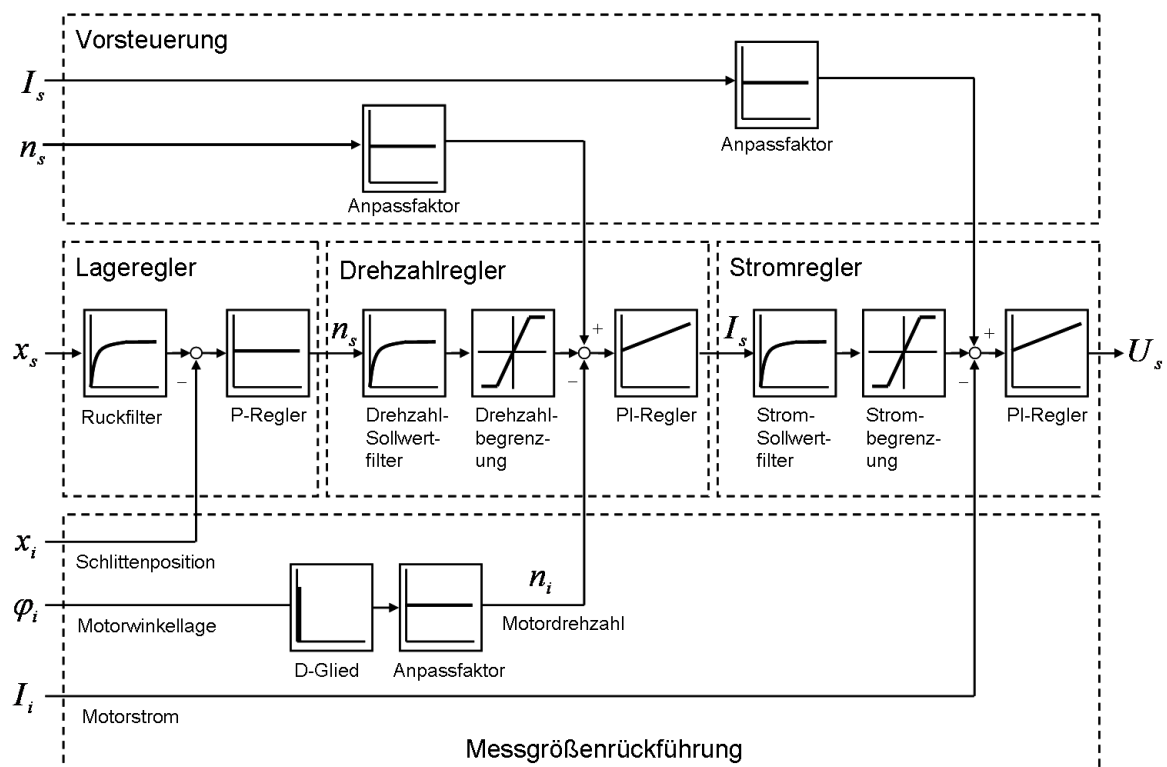


Abbildung 6.5: Kaskadierte Lageregelung für Vorschubantriebe von Werkzeugmaschinen nach [160]

und den entsprechenden Istwerten x_i des Positionsmesssystems die Drehzahlsollwerte n_s für den auf der eine Kaskadenebene darunter liegenden Drehzahlregler. Dieser berechnet wiederum nach einer Differenzbildung mit den gemessenen Motordrehzahlen n_i die Stromsollwerte I_s für den eine Kaskadenebene darunter liegenden Stromregler. Dieser gibt dann in Abhängigkeit der Soll-Ist-Differenz die Stellgrößen für das Antriebssystem aus.

Der Lageregler wird in Vorschubantrieben von Werkzeugmaschinen üblicherweise als Proportionalregler (P-Regler) ausgeführt. Dabei handelt es sich um einen Regler, der verzögerungsfrei auf eine Regeldifferenz reagiert und den Drehzahlstellwert über den Geschwindigkeitsverstärkungsfaktor K_V , der Proportionalbeiwert genannt wird, und einen Umrechnungsfaktor, bei dem es sich um eine Übersetzungskonstante des mechanischen Übertragungssystem handelt, anpasst. Der P-Regler besitzt keine Nachstelleigenschaften, wodurch er im Ergebnis eine bleibende Regeldifferenz aufweist. Dies äußert sich bei Betrachtung einer einzelnen Vorschubachse in einem Schleppfehler. Wird eine interpolierte, mehrachsige Bewegung betrachtet, äußert sich dies in einem Bahnfehler. Durch eine Anhebung des Geschwindigkeitsverstärkungsfaktors ist eine Minimierung dieses Fehlers erreichbar. Durch Erhöhung des Proportionalbeiwertes wird eine Erweiterung der Reglerbandbreite bewirkt, wodurch sich eine verbesserte dynamische Genauigkeit bei höheren Geschwindigkeiten ergibt. Allerdings entdämpft ein hoher Reglerbeiwert den Regelkreis, so dass der erreichbaren Dynamik eine Grenze gesetzt ist. Es ist üblich, dass der K_V -Wert des Lagereglers gerade so hoch eingestellt wird, dass in keiner Betriebssituation ein Überschwingen auftreten kann.

Zur Drehzahl- und Stromregelung wird im kaskadierten Lageregelkreis jeweils ein Regler mit Proportional- und Integralanteil (PI-Regler) verwendet. Bei diesem Regler bestimmt der Proportionalbeiwert K_P die Dynamik, während der Integralanteil gewährleistet, dass die Regeldifferenz vollständig zu Null geregelt wird. Eine kurze Nachstellzeit T_N bewirkt eine schnelle Kompensation von Störungen, allerdings auch eine Entdämpfung des Regelkreises. Hinsichtlich der Reglereinstellung lässt sich der Stromregler unabhängig von den Maschineneigenschaften parametrisieren, da sich nur der elektrische Teil des Antriebes im Stromregelkreis befindet. Der Drehzahlregler verarbeitet dagegen am mechanischen System gemessene Größen. In den Drehgebersignalen sichtbare Störbewegungen (Schwingungen) der Maschinenstruktur werden daher über den Drehzahlregler rückgekoppelt und begrenzen dessen Proportionalbeiwert.

Zur regelungstechnischen Bedämpfung von mechanischen Resonanzen werden Führungsgrößenfilter eingesetzt. Durch die selektive Filterung kritischer Frequenzbänder wird deren Anregung unterdrückt und eine weitere Anhebung der Reglerbeiwerte möglich. Ein Ruckfilter am Eingang des Lagereglers glättet bereits die vom Interpolator abgegebenen Führungsgrößen und reduziert so die Anregung mechanischer Eigenfrequenzen durch zu hohen Ruck [1].

Zum Schutz der Leistung führenden Bauteile vor elektrischer oder mechanischer Überlastung dienen Begrenzungsglieder in den Regelkreisen.

6.4 Dynamisches Verhalten

Die Entwicklungen im Bereich der Prozesstechnik haben drastische Steigerungen der Leistung von Werkzeugmaschinen ermöglicht. Besonders im Bereich der spanenden Werkzeugmaschinen ermöglichte diese Entwicklung eine drastische Steigerung der Zerspanleistung. Die Zerspanung mit hohen Leistungen erfordert eine steife Auslegung der Maschinenkonstruktion. Bei Schwingungen der Schneide wird eine wellige Oberfläche erzeugt, die bei entsprechender Phasenlage der nachfolgenden Schneide zu einer dynamischen Modulation der Schnitttiefen und damit der Spanungsdicken und damit einhergehend auch mit den Zerspankräften führt. Die Rückkopplung der modulierten Zerspankraftanteile auf die Maschine kann zur Entdämpfung der Schwingungen bis hin zur Instabilität führen. Ein instabiles Verhalten der Maschine macht sich akustisch durch Rattern und am Werkstück durch Rattermarken bemerkbar. Im Extremfall führen die durch Instabilität verursachten Kräfte zu Schäden am Werkzeug oder an der Maschine.

Eine Erhöhung der Schnittgeschwindigkeiten moderner spanender Werkzeugmaschinen lässt sich auf begrenzten Verfahrenswegen nur mit hohen Beschleunigungen und hohem Ruck realisieren. Damit ergeben sich hinsichtlich der Bewegungsdynamik neue Anforderungen und veränderte Auslegungskriterien der Werkzeugmaschinen, da sich durch schnelle Positioniervorgänge mit hoher Geschwindigkeit und hohem Ruck statische und dynamische Belastungen der Werkzeugmaschine ergeben.

Um höhere Beschleunigungen zu erreichen, muss die bewegte Masse reduziert werden. Mit der weiterhin bestehenden Steifigkeitsanforderung ergibt sich für die Konstruktion ein Verbund konkurrierender Teilziele, der durch Kompromisse gelöst werden muss. Als optimale Lösung dieses Problems bieten sich Leichtbaukonstruktionen an, die die geforderte hohe Steifigkeit bei minima-

ler Masse erreichen. Der benötigte hohe Ruck führt zur Erregung von Eigenfrequenzen in einem breiten Frequenzband. Da Schwingungen die Stabilität der Regelung beeinflussen, müssen die Konstruktionen auch hinsichtlich ihrer dynamischen Übertragungseigenschaften im Regelkreis optimiert werden.

Bei heutigen Werkzeugmaschinen werden bestimmte Eigenfrequenzen durch strukturelastisches Verhalten sowohl der Antriebsstränge als auch des Maschinengestells verursacht. Aus diesem Grund stellt die optimale strukturmechanische Auslegung des Maschinengestells und der Antriebsstränge sowie der regelungstechnischen Eigenschaften des mechatronischen Gesamtsystems auf Grund der vielen Einflussfaktoren eine komplexe Entwicklungsaufgabe dar, die nur mit Hilfe von numerischen Simulationsverfahren gelöst werden kann.

Kapitel 7

Simulation von Werkzeugmaschinen

Eine Möglichkeit die Genauigkeit des dynamischen Verhaltens von Werkzeugmaschinen in gewissen Grenzen zu verbessern, ergibt sich durch die Verwendung erweiterter Regelalgorithmen. Das Verhalten der Regelung wird dabei mit Filtern und bzw. oder aktiver Dämpfung optimal an die Eigenschaften der Regelstrecke angepasst [1].

Um das maximale Leistungspotential einer Werkzeugmaschine auszuschöpfen, ist allerdings die optimale Auslegung der mechanischen Konstruktion im Hinblick auf ihr Zusammenspiel mit der Antriebsregelung entscheidend. Auf Grund der engen Integration mechanischer, elektro- und informationstechnischer Funktionsträger in Aufbau, Funktionsweise und Verhalten stellen diese Auslegungsaufgaben die Produktentwicklung vor große Herausforderungen. Um diesen Auslegungprozess zu unterstützen, ist eine gesamtheitliche Simulation der Werkzeugmaschine als mechatronisches System erforderlich. Aus diesem Grund werden in diesem Kapitel die grundlegenden Vorgehensweisen bei der Simulation von Werkzeugmaschinen beschrieben.

Die Simulation und regelungstechnische Analyse der Antriebssysteme erfolgt oft in einer getrennten Betrachtung. Auf Grund der bekannten dynamischen Wechselwirkungen der Teilsysteme ist allerdings oft eine gesamtheitliche Simulation der Werkzeugmaschinen als mechatronische Systeme erwünscht. Hierfür ist die Computersimulation als hilfreiches Mittel zu sehen. Die großen Leistungszuwächse im Bereich der Hard- und Software in den letzten Jahren ermöglichen heute den wirtschaftlichen Einsatz numerischer Berechnungswerkzeuge, welche die Simulation komplexer technischer Systeme anhand von digitalen Modellen erlauben. Damit können Entwürfe von Maschinen bereits vor der Investition in Prototypen analysiert und optimiert werden.

Dieses Kapitel stellt die Grundlagen der Simulation von Werkzeugmaschinen dar. Das darin dargestellte Themengebiet steht nicht im Fokus der vorliegenden Arbeit, vielmehr sollen die Darstellungen zum Verständnis der nachfolgenden Kapitel beitragen. Die Darstellungen dieses Kapitels stützen sich dabei auf [138].

In Abschnitt 7.1 wird die grundlegende Vorgehensweise zur strukturmechanischen Berechnung der Regelsimulation von Werkzeugmaschinen auf Basis eines integrierten Computermodells der Antriebsstränge, der elektrischen Antriebe und der Antriebsregelung beschrieben. In den darauf folgenden Abschnitten 7.2 und 7.3 wird die Modellbildung der Regelung und der elektrischen Antriebe sowie die verschiedenen Ansätze zur Modellbildung mechanischer Strukturen dargestellt.

In diesem Zusammenhang wird auf die Modellierung der Gestell- und Antriebsstrukturen mit der Finiten-Elemente-Methode näher eingegangen. In Abschnitt 7.4 wird die Modellordnungsreduktion als Anwendung in der Simulation von Werkzeugmaschinen betrachtet. Im abschließenden Abschnitt 7.5 erfolgt die Darstellung einer Kopplung des regelungs- und elektrotechnischen Modells mit dem Modell der mechanischen Struktur einer Werkzeugmaschine zu einem mechatronischen Gesamtsystems.

7.1 Regelungssimulation von Werkzeugmaschinen

Werkzeugmaschinen sind in ihrer Ausprägung als mechatronische Systeme komplexe Systeme mit heterogener physikalischer Beschreibungsform. Für die Modellbildung und Simulation derartiger Systeme existiert bis heute keine einheitliche und allgemeingültige Systematik. Eine in der Praxis übliche Vorgehensweise besteht in der Zerlegung des betrachteten heterogenen Systems in Teilsysteme, deren Verhaltensweisen jeweils separat mathematisch geschlossen beschreib- und lösbar sind. Die Teilsysteme besitzen dabei Randbedingungen (Kräfte, Verschiebungen, Spannungen, Ströme), die durch das Verhalten der jeweils benachbarten Teilsysteme vorgegeben werden. Durch Einführung geeigneter Koppelbedingungen kann eine Lösung des Gesamtsystems berechnet werden. In der praktischen Anwendung erfolgt die Formulierung, Kopplung und die Berechnung einer Lösung der Teilsysteme entweder in einer einzigen Simulationsumgebung oder durch die Kopplung der Solver¹ verschiedener Simulationssysteme in einer so genannten Co-Simulation.

Eine Werkzeugmaschine kann mit den Eigenschaften eines mechatronischen Systems als regelungstechnisches System betrachtet werden. In den folgenden Abschnitten wird die für regelungstechnische Systeme übliche Darstellung der Wirkstruktur durch Blockschaltbilder dargestellt.

7.1.1 Regelungstechnische Systeme

Der grundsätzliche Aufbau eines Regelkreises besteht aus der Regelung, dem Stellglied und der Regelstrecke. Wird eine Werkzeugmaschine als Regelkreis aufgefasst, ergibt sich durch diese Aufteilung ein informationstechnisches, ein elektrotechnisches und ein mechanisches Teilsystem, wie in Abbildung 7.1 dargestellt. Für die Simulation benötigte Modellbildung des Regelkreises bietet sich die Beschreibung der informations- und elektrotechnischen Teile durch regelungstechnische Darstellungsformen und der Regelstrecke mittels mechanischer Verfahren an.

Die Methoden der Mechanik beschreiben dabei die Bewegung fester Körper unter Zwangsbedingungen und der Einwirkung von äußeren Kräften, während die Algorithmen der Regelungstechnik die Kräfte definieren, die erforderlich sind, um einen Körper mit Hilfe des Stellgliedes eine gewünschte Bewegung aufzuprägen.

Zur Modellierung und zur Darstellung von Regelungssystemen sind Blockschaltbilder, wie sie in Abbildung 7.1 verwendet wurden, üblich. Ein Modellblock bildet dabei das Übertragungsverhalten eines Teilsystems durch eine Gleichung oder ein Gleichungssystem ab. Die einzelnen

¹Englisch: Gleichungslöser

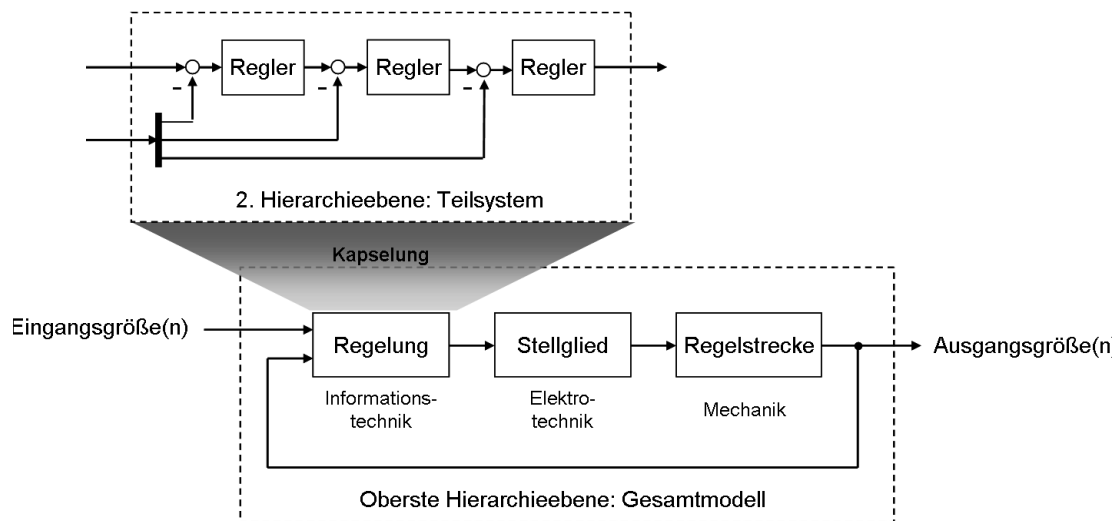


Abbildung 7.1: Blockorientierte Modellierung nach [63]

Modellblöcke werden im Regelkreis so angeordnet, dass die Ausgangsgröße eines Teilsystems der Eingangsgröße des nachfolgenden Teilsystems entspricht. Eine Kopplung der Gleichungen bzw. der Gleichungssysteme wird durch die signaltechnische Verknüpfung der entsprechenden Ein- und Ausgangsgrößen erreicht [63].

Mit Hilfe der elementaren Blockschaltbilder lassen sich somit Wirkstrukturen mit beliebiger Komplexität und vielschichtiger Verknüpfung modellieren. Nichtlineare Gleichungssysteme können direkt in geschlossener Form aufgestellt, linearisiert und im Zeit- oder Frequenzbereich gelöst werden. Auf diese Weise ist es möglich, sowohl lineare als auch nichtlineare Systeme zu modellieren. Dabei werden im Falle einer Simulation von nichtlinearen Verhaltensweisen im Zeitbereich die Gleichungen der einzelnen Modellblöcke durch Zeitbereichsintegration separat gelöst und die Randbedingungen an diskreten Zeitpunkten ausgetauscht.

Moderne Simulationsprogramme wie z.B. MATLAB/Simulink[®] basieren auf grafisch blockorientierten Editoren für die Modellerstellung. Diese ermöglichen neben einer relativ leicht handhabbaren Modellierung durch Kapselung von Teilmodellen auch eine hierarchische Strukturierung der Modelle, womit selbst komplexe Systeme in übersichtlicher Form beschrieben werden können. Einzelne Modellblöcke können demnach beliebig komplexe Algorithmen enthalten, auf der obersten Reglerhierarchie ergibt sich dennoch ein übersichtlicher Regelkreis. In Abbildung 7.1 wird dies am Beispiel eines mechatronischen Systems verdeutlicht.

Grundsätzlich orientiert sich die Beschreibung der Modellblöcke an den spezifischen Gegebenheiten der Teilsysteme. Da bei Werkzeugmaschinen die antriebstechnische Lösung mit kaskadierter Lageregelung verbreitet ist, sind die Wirkstrukturen der Regelungen und der Stelleinrichtungen weitestgehend festgelegt, siehe Abschnitt 6.3. Aus diesem Grund kann die Modellbildung der informations- und der elektrotechnischen Teilsysteme einmalig konfiguriert werden. Zur Simulation einer Maschine ist dann lediglich eine Parametrisierung nötig, die an das Verhalten der Regelstrecke angepasst wird.

7.1.2 Grundstruktur des mechatronischen Systems

Wird eine Werkzeugmaschine als mechatronisches Gesamtsystem modelliert, wird die für regelungstechnische Systeme übliche Darstellung der Wirkstruktur durch Blockschaltbilder verwendet. Dabei wird das dynamische Verhalten der dabei betrachteten Systeme im wesentlichen durch die Übertragungseigenschaften der Lageregelung, der elektrischen Antriebe und des mechanischen Systems sowie durch die Wechselwirkungen dieser Teilsysteme im Regelkreis bewirkt. Für die Modellbildung einer Werkzeugmaschine als mechatronisches Gesamtsystems wird dieses in Teilsysteme der Achsen und des Gestells sowie der Antriebsstränge zerlegt und ein Blockschaltbild, wie es in Abbildung 7.2 dargestellt ist, als Grundstruktur des Modells definiert. Der

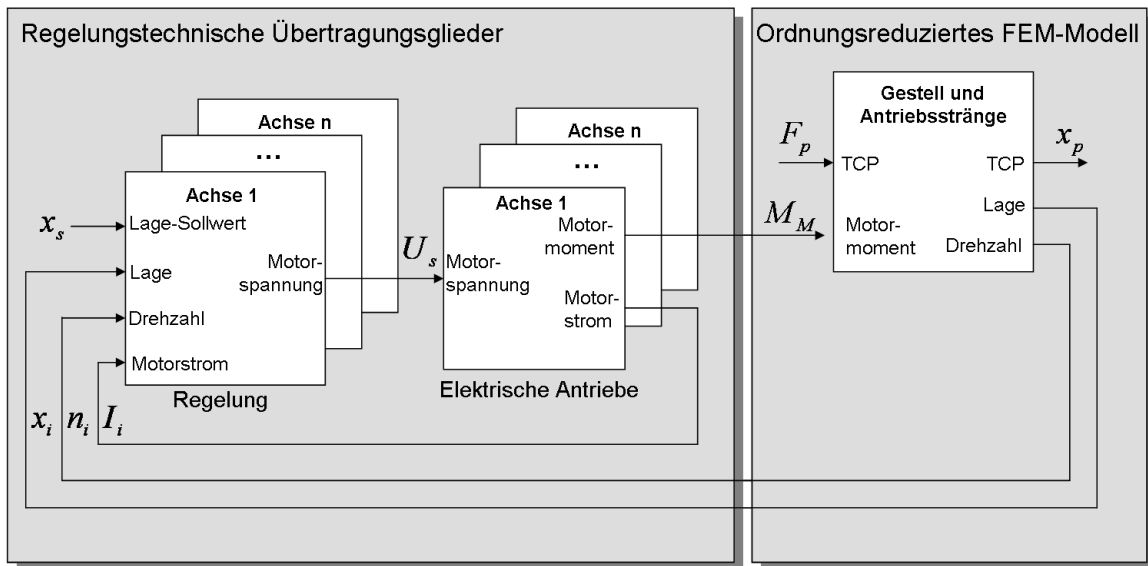


Abbildung 7.2: Mechatronisches Gesamtmodell nach [138]

Modellblock der Regelung berechnet mit der am Eingang anliegenden Soll- und Istwerte der Lagen x , der Drehzahlen n und der Motorströme I die Sollwerte U_s der zur Ansteuerung der elektrischen Antriebe notwendigen Motorspannungen. Das nachfolgende Modell der elektrischen Antriebe ermittelt daraus die Motormomente M_M , die dann am Eingang des mechanischen Systems anliegen. Die zur Erzeugung der Motormomente notwendigen Motorströme I_i werden an den Eingang der Regelung zurückgegeben.

Die Aufgabe des mechanischen Modells besteht in der Simulation der durch die Motormomente hervorgerufenen Bewegungen der Maschine. Die mit Hilfe der simulierten Bewegungen der Maschine berechneten Schlittenlagen und Motordrehzahlen werden an die Regelung zurückgeführt. Über eine Störgrößenschnittstelle können dabei auch Störkräfte F_P , wie zum Beispiel Zerspankräfte am *Tool Center Point* (TCP), eingeleitet und die resultierenden Bewegungen x_P der Störfreiheitsgrade, wie zum Beispiel Relativbewegungen zwischen Werkzeug und Werkstück, ausgegeben werden. Bei einer vektoriellen Darstellung beschreiben die Modellblöcke der Regelung und der elektrischen Antriebe das Verhalten beliebig vieler gleich strukturierter Achsen, in diesem Fall sind die ausgetauschten Signale ebenfalls Vektoren.

Das in Abbildung 7.2 dargestellte Blockschaltbild stellt die oberste Hierarchieebene des mechatronischen Gesamtsystems dar. In den folgenden Abschnitten wird auf die Modellbildung der Teilsysteme detaillierter eingegangen. Die Wirkstrukturen der Regelung und der elektrischen Antriebe können mit elementaren Übertragungsgliedern modelliert werden. Im Abschnitt 7.2 wird dies näher dargestellt. Das Modell des mechanischen Systems muss das Schwingungsverhalten der Maschine möglichst genau abbilden und alle Effekte, welche die Reglerdynamik beeinflussen, enthalten. Zu diesem Zweck wird in Abschnitt 7.3 eine Methode vorgestellt, mit der geeignete FEM-Modelle generiert werden können. Diese beschreiben das Strukturverhalten sowohl des Maschinengestells als auch der Antriebsstränge detailliert und in geschlossener Form. Da diese Modelle einen hohen Detaillierungsgrad aufweisen, müssen diese in der Ordnung reduziert werden, um akzeptable Rechenzeiten zu ermöglichen. Das ordnungsreduzierte Gleichungssystem des FEM-Modells kann anschließend in das mechatronische Gesamtmodell integriert werden, wodurch eine Kopplung des regelungs/elektrotechnischen und des mechanischen Gleichungssystems erzielt wird.

Auf Basis des mechatronischen Gesamtmodells kann daraufhin das Verhalten der Vorschubantriebe innerhalb der Maschine simuliert und nach regelungstechnischen Gesichtspunkten im Zeit- und Frequenzbereich analysiert werden.

7.2 Modellbildung der Regelung und der elektrischen Antriebe

Wie bereits im vorhergehenden Abschnitt 6.3 erläutert, wird zur Regelung der Antriebsstränge einer Werkzeugmaschine kaskadierte Regelkreise verwendet. In diesem Abschnitt erfolgt die Darstellung der Modellbildung solcher Regelkreise sowie der elektrischen Antriebe mit der für regelungstechnische Systeme üblichen Darstellung durch Blockschaltbilder.

7.2.1 Modellbildung der Regelung

Das Modell der Regelung dient der Abbildung des für das dynamische Verhalten der Maschine wesentlichen Aufbaus der eingesetzten Antriebsregelung. In Abschnitt 6.3 wurde bereits dargestellt, dass bei Werkzeugmaschinen eine Kaskadenregelung üblich ist. Die Abbildung 7.3 zeigt ein auf viele Anwendungsfälle übertragbares Reglermodell. Dieses kann, wie dargestellt, neben den Regelgliedern auch Sollwertfilter berücksichtigen. Wie aus der Darstellung des Modells ersichtlich muss das Modell primär den proportional verstärkenden Lageregler (P-Regler) und die unterlagerten Drehzahl- und Stromregler mit Proportional- und Integralanteil (PI-Regler) beinhalten.

7.2.1.1 P- und PI-Regler

Die Abbildung 7.4 zeigt das Blockschaltbild des PI-Reglers. Dieser Regler zeichnet sich durch ein verzögerungsfreies Ansprechen des Proportionalanteils (P-Anteil), der von den Parametern K_P und K_A abhängt, auf die Regeldifferenz (Eingangsgröße u) aus, während der Integralanteil (I-Anteil) gewährleistet, dass diese in einer endlichen Zeit, die vom Parameter T_N abhängt,

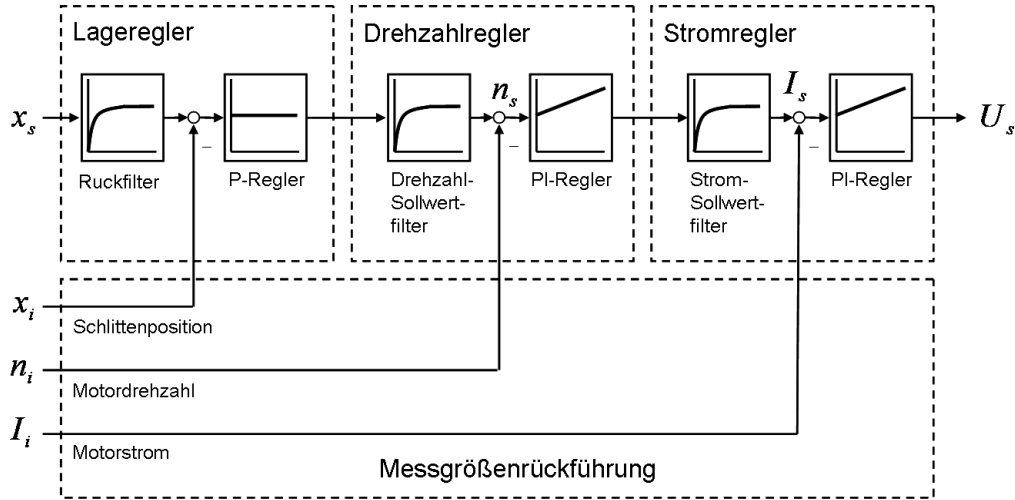


Abbildung 7.3: Vereinfachtes Modell einer kaskadierten Lageregelung nach [160]

zu Null geregelt wird. Der Parameter K_P wird Proportionalbeiwert, der Parameter K_A wird Anpassungsfaktor und der Parameter T_N Nachstellzeit genannt. Die Normierung des Proportionalbeiwertes K_P durch einen Anpassungsfaktor K_A ist üblich, um die Reglerparameter unterschiedlicher Antriebskonfigurationen, z.B. mit verschiedenen Getriebeübersetzungen, Spindelsteigungen, Drehmomentkonstanten oder Massen, vergleichbar zu machen. Der Integrator ($1/s$)

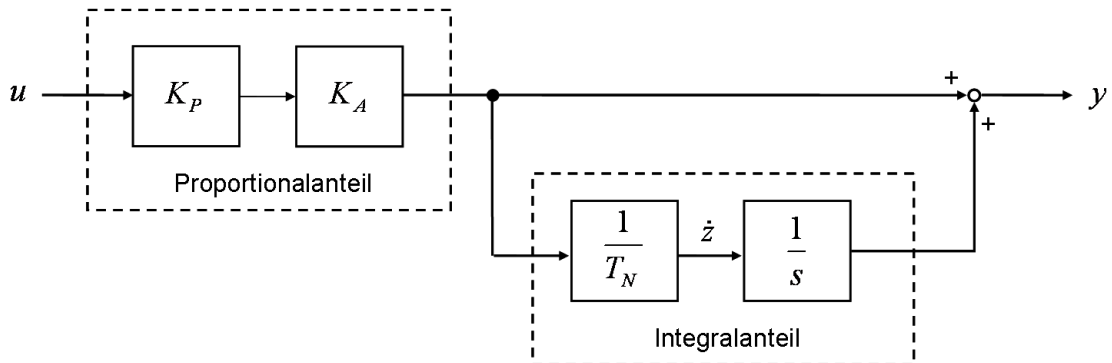


Abbildung 7.4: Blockschaltbild des PI-Reglers nach [63]

erzeugt eine Zustandsvariable z im Gleichungssystem der Regelung. Das Verhalten des PI-Reglers mit der Eingangsgröße u (der Regeldifferenz) und der Ausgangsgröße y (Stellgröße) kann in der Form einer Zustandsdifferenzial- und einer Ausgangsgleichung angegeben werden:

$$\dot{z} = \frac{K_P K_A}{T_N} \cdot u \quad (7.1)$$

$$y = z + K_P K_A \cdot u. \quad (7.2)$$

PI-Regler werden bei Werkzeugmaschinen nur in den unterlagerten Regelkreisen zur Regelung der Drehzahlen und der Motorströme eingesetzt, da durch den Integralanteil die Gefahr des Überschwingens besteht. Bei der Positionsregelung muss dieses Verhalten aus Gründen der Arbeitsgenauigkeit ausgeschlossen werden. Deshalb wird die Lageregelung als Regelung mit einem reinen P-Regler ausgeführt. Die Gleichung dieses Reglers beschränkt sich deshalb auf den

proportionalen Zusammenhang zwischen der Eingangsgröße u (Lagedifferenz) und y (Drehzahl-Sollwert):

$$y = K_V K_A \cdot u. \quad (7.3)$$

Hierbei wird für den Proportionalbeiwert des Lagereglers die bei Werkzeugmaschinen gebräuchliche Bezeichnung K_V verwendet, der in diesem Fall Geschwindigkeitsverstärkungsfaktor genannt wird. Der Anpassungsfaktor K_A rechnet in diesem Fall den Geschwindigkeits-Sollwert in eine Motordrehzahl um.

7.2.1.2 Führungsgrößenfilter

Die Berücksichtigung von Filtern im Rahmen einer Simulation orientiert sich am Anwendungsfall. Bei bereits bestehenden Systemen, die konstruktiv nicht mehr angepasst werden können, müssen sie berücksichtigt werden, um das gegebene Übertragungsverhalten realistisch abzubilden. Zur Vorherbestimmung und Optimierung von mechatronischen Systemeigenschaften in der Konstruktionsphase sind Filter weniger zweckmäßig, da Filter auf das Verhalten des mechanischen Systems angepasst sind und Resonanzphänomene und die Übertragungseigenschaften somit vor dem Filterentwurf analysiert werden müssen. Da Filter das Leistungspotential von Systementwürfen nicht erweitern, sollten mechanische Resonanzen zunächst durch konstruktive Maßnahmen behandelt werden.

In der Signalverarbeitung werden verschiedene Filtertypen verwendet. Hier seien nur die Tiefpassfilter mit PT1- oder PT2-Verhalten betrachtet, die in den Antriebsregelkreisen von Werkzeugmaschinen sehr häufig zum Einsatz kommen, um hochfrequente Störungen zu unterdrücken. In der Abbildung 7.5 sind die Blockschaltbilder dieser Filter dargestellt. Das Verhalten des PT1-

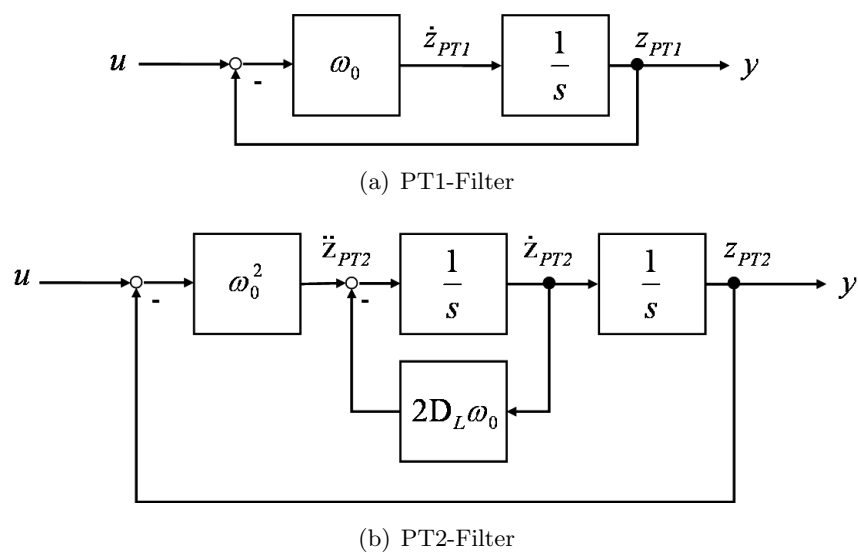


Abbildung 7.5: Tiefpassfilter nach [63]

Filters mit der Grenzfrequenz ω_0 der Eingangsgröße u und der Ausgangsgröße y wird durch eine

Differenzialgleichung erster Ordnung beschrieben:

$$\dot{z}_{PT1} = (u - z_{PT1}) \cdot \omega_0 \quad (7.4)$$

$$y = z_{PT1}. \quad (7.5)$$

Nach einer Laplace-Transformation erhält man die Übertragungsfunktion H_{PT1} des PT1-Filters:

$$H_{PT1}(s) = \frac{\omega_0}{s + \omega_0}. \quad (7.6)$$

PT2-Filter mit der Grenzfrequenz ω_0 und dem Dämpfungsmaß D werden durch Differenzialgleichungen zweiter Ordnung beschrieben. Mit den Zustandsvariablen \dot{z}_{PT2} und z_{PT2} können diese in Zustandsraumdarstellung dargestellt werden:

$$\begin{bmatrix} \ddot{z}_{PT2} \\ \dot{z}_{PT2} \end{bmatrix} = \begin{bmatrix} -2D\omega_0 & -\omega_0^2 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{z}_{PT2} \\ z_{PT2} \end{bmatrix} + \begin{bmatrix} \omega_0^2 \\ 0 \end{bmatrix} u \quad (7.7)$$

$$y = [0 \ 1] \begin{bmatrix} \dot{z}_{PT2} \\ z_{PT2} \end{bmatrix}. \quad (7.8)$$

Wie die Überführung in den Bildbereich zeigt, hat die Übertragungsfunktion des PT2-Filters die folgende Form:

$$H_{PT2}(s) = \frac{\omega_0^2}{s + 2D\omega_0 s + \omega_0^2}. \quad (7.9)$$

7.2.2 Modellbildung der elektrischen Antriebe

Die Darstellung der Modellbildung der elektrischen Antriebe erfolgt hier anhand des bei Werkzeugmaschinen überwiegend verwendeten Servomotors und anhand eines Transistorstellers.

7.2.2.1 Servomotor

Der Servomotor dient der Umwandlung von elektrischer Energie in ein Drehmoment, das an der Motorwelle abgegeben wird. In Werkzeugmaschinen werden heute überwiegend feldorientierte und geregelte Synchron-Drehstrommotoren eingesetzt. Die mechanischen Eigenschaften des Motors werden dem mechanischen Modell zugeordnet. Deshalb ist beim Aufstellen des Modells des Servomotors lediglich noch das elektrodynamische Übertragungsverhalten mit der angelegten Motorspannung als Eingangsgröße und dem Motormoment als Ausgangsgröße zu berücksichtigen. Der Motorstrom ist als Regelgröße an den Eingang des Stromreglers zurückzuführen und wird daher als zweite Ausgangsgröße ausgeleitet.

Anhand einer blockorientierten Darstellung kann das Ein- und Ausgabeverhalten prinzipiell mit dem Modell der einer Gleichstrommaschine angenähert werden. Dabei sind die Parameter so zu wählen, dass das zeitliche Verhalten der Ersatzschaltung durch den Gleichstrommotor dem der am Servo-Umrichter betriebenen permanentenerregten Synchronmaschine gleicht. In Abbildung 7.6 sind die elektrischen Ersatzschaltbilder für die Ankerwicklungen der Synchronmaschine und der Gleichstrom-Nebenschlussmaschine dargestellt. Die Motorspannung U_A liegt am Widerstand R und an der Induktivität L der Spulenwicklungen an. Unter der Vernachlässigung der induzierten

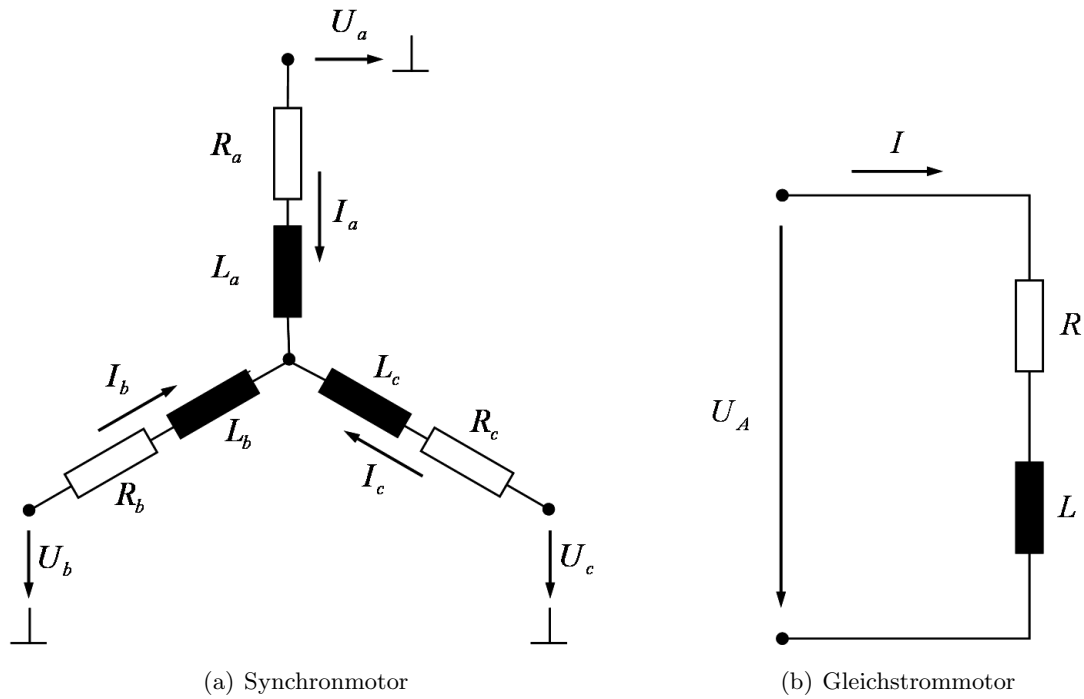


Abbildung 7.6: Elektrische Ersatzschaltbilder des Synchron- und des Gleichstrommotors

Gegenspannung, die im linearen Betrieb des Stromregelkreises im Vergleich zur Soll-Spannung sehr klein ist und sich innerhalb der Stromanstiegs- und Stromabfallzeiten nur unwesentlich ändert, gilt für die Spannungen im Ankerkreis:

$$U_A = R \cdot I + L \cdot \dot{I}. \quad (7.10)$$

Mit dieser Gleichung in der Form einer Zustandsdifferenzialgleichung und einer Ausgangsgleichung, wobei der Motorstrom I einer Zustandsvariable z_{SM} entspricht, erhält man:

$$\dot{z}_{SM} = \frac{1}{L} \cdot U_A - \frac{R}{L} \cdot z_{SM} \quad (7.11)$$

$$I = z_{SM}. \quad (7.12)$$

Das Motormoment M_M wird durch eine zweite Ausgangsgleichung angegeben:

$$M_M = K_M \cdot z_{SM}, \quad (7.13)$$

wobei K_M die Drehmomentkonstante des Motors genannt wird. Die Abbildung 7.7 stellt das Blockschaltbild eines Servomotors dar. Mit dieser Modellierung liegt für den Servomotor ein einfaches Ersatzmodell mit einfachem PT1-Verhalten vor, dessen Übertragungsfunktion H_M man durch die Laplace-Transformation der Gleichung (7.10) erhält:

$$H_M(s) = \frac{1/R}{1 + (L/R) \cdot s}. \quad (7.14)$$

7.2.2.2 Transistorsteller

Die Aufgabe des Transistorstellers ist die als Gleichspannung zwischengespeicherte elektrische Energie gemäß der vom Stromregler berechneten Spannungsvorgaben in pulsweitenmodulier-

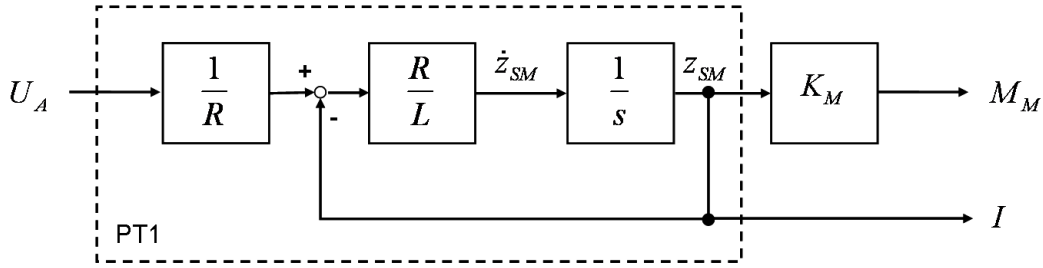


Abbildung 7.7: Blockschaltbild eines Servomotors nach [138]

ter Form auf die Wicklungen des Servomotors zu schalten, um einen geglätteten sinusförmigen Verlauf der Phasenströme zu erhalten. Das dynamische Verhalten des Transistorstellers hängt im wesentlichen von den Verzögerungszeiten ab, die durch die Schaltzeiten der Transistoren zuzüglich der durch die digitalen Signalprozessoren zur Berechnung der Schaltzeitpunkte benötigten Rechenzeit ergeben [33].

Aus der regelungstechnischen Sichtweise stellt der Transistorsteller wegen der zeitlichen Verschiebung des Ausgangssignals näherungsweise ein Totzeitglied dar, siehe [31, 120]. Die Totzeit T_T resultiert aus der gerätetechnischen Rechenzeit T_R sowie aus der halben Periodendauer des Transistorstellers, die sich aus der Schaltfrequenz f_W und der Rechenzeit p_T ergibt. Die Pulszahl gibt an, wie viele Strompulse in einer Schaltperiode abgegeben werden [163, 183]:

$$T_T = T_R + \frac{1}{2 \cdot f_W \cdot p_T}. \quad (7.15)$$

Aus systemtechnischer Sicht stellt der Transistorsteller ein nichtlineares Übertragungsglied dar. Für lineare Analysen im Frequenzbereich kann das Totzeitglied durch ein Verzögerungsglied erster Ordnung approximiert werden [56]. In Abbildung 7.8 ist ein solches Verzögerungsglied dargestellt. Mit der Zustandsvariablen z_{TR} für die Ankerspannung U_A lautet die Gleichung des

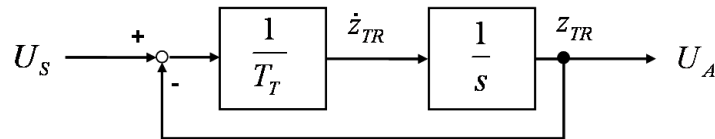


Abbildung 7.8: Blockschaltbild des Transistorstellers mit PT1-Verhalten nach [138]

Transistorstellers in Zustandsraumdarstellung:

$$\dot{z}_{TR} = \frac{1}{T_T}(U_s - z_{TR}) \quad (7.16)$$

$$U_A = z_{TR}. \quad (7.17)$$

7.2.3 Gleichungssystem der Regelung und der elektrischen Antriebe

In diesem Abschnitt wird der aus der Modellbildung der Regelung und der Modellbildung der elektrischen Antriebe hervorgehende Teil des mechatronischen Systems (Abbildung 7.9) und des daraus resultierenden Gleichungssystems betrachtet. Die Blockschaltglieder, die in den Ab-

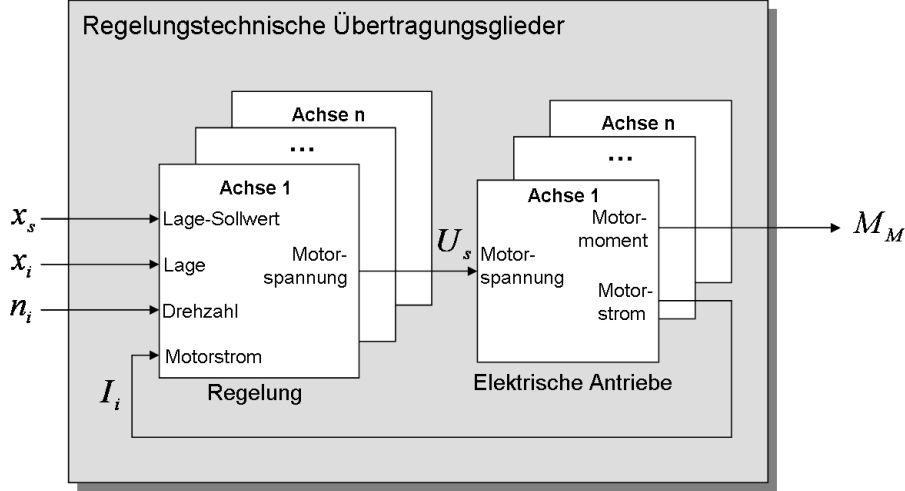


Abbildung 7.9: Regelungs- und elektrotechnischer Teil eines mechatronischen Gesamtsystems nach [138]

schnitten 7.2.1 und 7.2.2 beschrieben wurden, werden durch Differenzialgleichungen beschrieben. Durch die Verknüpfung der Glieder wird ein Differenzialgleichungssystem definiert:

$$\begin{aligned} \dot{\mathbf{z}}_R &= \mathbf{A}_R \mathbf{z}_R + [\mathbf{B}_{R,s} \quad -\mathbf{B}_{R,i}] \begin{bmatrix} \mathbf{u}_{R,s} \\ \mathbf{u}_{R,i} \end{bmatrix} \\ \mathbf{y}_R &= \mathbf{C}_R \mathbf{z}_R, \end{aligned} \quad (7.18)$$

wobei \mathbf{A}_R die Systemmatrix und \mathbf{z}_R den Vektor der regelungs- und der elektrotechnischen Systemzustände darstellt. Die Zustandsvariablen, die durch die Integratoren der P- und PI-Regler sowie der PT1- und PT2-Glieder erzeugt werden, werden im Zustandsvektor \mathbf{z}_R zusammengefasst. Für ein einfaches Antriebssystem mit Kaskadenregelung sind dies die Zustände der Drehzahlregler \mathbf{z}_{DR} , der Stromregler \mathbf{z}_{SR} , der Transistorsteller \mathbf{z}_{TR} und der Servomotoren \mathbf{z}_{SM} :

$$\dot{\mathbf{z}}_R := [\mathbf{z}_{DR}^T \quad \mathbf{z}_{SR}^T \quad \mathbf{z}_{TR}^T \quad \mathbf{z}_{SM}^T]^T. \quad (7.19)$$

Werden weitere integrierende Glieder, wie zum Beispiel Filter, im Modell verwendet, kommen entsprechend weitere Zustandsvariablen hinzu. Am Eingang der Regelung liegen die Soll- (\mathbf{x}_s) und Istwerte ($\mathbf{x}_i, \mathbf{n}_i$) der Lagen und Geschwindigkeiten an. Zur übersichtlichen Darstellung der Messgrößenrückführung bei der Kopplung mit dem mechanischen System werden die Soll- und Istwerte über separierte Einzelmatrizen $\mathbf{B}_{R,s}$ und $\mathbf{B}_{R,i}$ getrennt eingeleitet. Entsprechend teilen sich auch die Eingangsgrößen auf zwei Eingangsvektoren $\mathbf{u}_{R,s}$ und $\mathbf{u}_{R,i}$ auf:

$$\mathbf{u}_{R,s} := [\mathbf{x}_s^T \quad \mathbf{n}_s^T \quad \mathbf{I}_s^T]^T \quad \text{und} \quad \mathbf{u}_{R,i} := [\mathbf{x}_i^T \quad \mathbf{n}_i^T]^T. \quad (7.20)$$

Die gegenüber den vorhergehenden Darstellungen zusätzlich eingeführten Eingänge für Drehzahl- und Stromsollwerte \mathbf{n}_s bzw. \mathbf{I}_s , ermöglichen Analysen der unterlagerten Regelkreise durch Deaktivierung des Lage- und/oder Drehzahlreglers. In Abbildung 7.10 ist ein Reglermodell mit

Schalter S_1 und S_2 zum Aktivieren und Deaktivieren vor den unterlagerten Reglern, dargestellt. Die Eingänge n_s und I_s dienen zudem zur Einleitung von Vorsteuergrößen, die über die Schalter V_n bzw. V_I an den jeweiligen Reglereingängen aufschaltbar sind. Der Ausgangsvektor

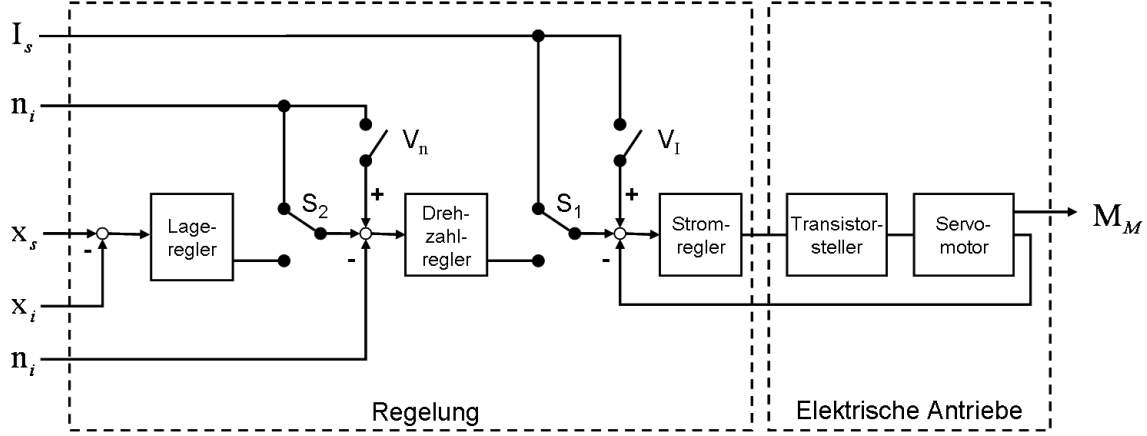


Abbildung 7.10: Reglermodell mit Schaltern S_1, S_2 zum De- und Aktivieren der Regelkreise sowie V_n und V_I zur Aufschaltung von Vorsteuergrößen nach [138]

y_R enthält die Motormomente M_M , die über die Ausgangsmatrix C_R ausgeleitet werden:

$$y_R := [M_M]. \quad (7.21)$$

Die Abbildung 7.11 zeigt die Zustandsraumdarstellung der regelungs- und elektrotechnischen Systemgleichungen (7.18) und (7.19) als Blockschaltbild.

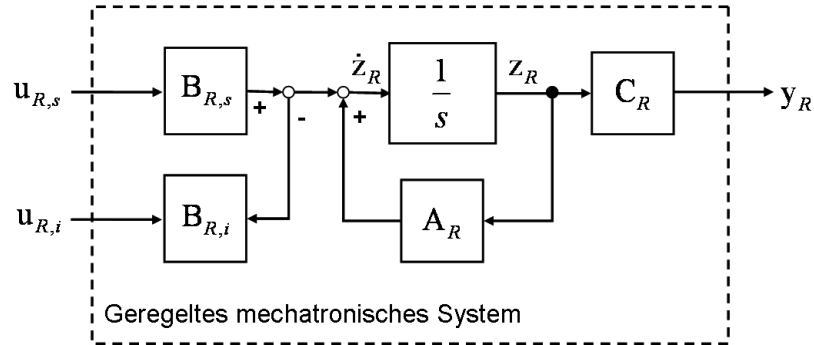


Abbildung 7.11: Blockschaltbild des regelungs- und des elektrotechnischen Gleichungssystems in Zustandsdarstellung nach [138]

7.3 Modellbildung des mechanischen Systems

7.3.1 Modellierungsansätze

Während die Modellbildung der informations- und der elektrotechnischen Teilsysteme weitgehend durch deren festgelegte Wirkstrukturen der Regelungen festgelegt sind, erfordert die Modellbildung des mechanischen Systems eine individuelle Betrachtung des konstruktiven Aufbaus

der betrachteten Maschine. Die Ursache dafür liegt in der Beeinflussung des Regelbetriebes durch die Eigenfrequenzen der Maschine, die durch die Struktur der Maschine bestimmt werden.

Bei der Beschreibung der Bewegung fester Körper ist zwischen Starrkörperbewegungen, bei denen es sich um Verschiebungen und Drehungen der Körper ohne innere Verzerrungen handelt und elastische und plastische Verformungen zu unterscheiden. Grundsätzlich unterliegen physikalische Körper immer einem Überlagerungszustand dieser Bewegungsanteile. Bei der Beschreibung des Verhaltens von Werkzeugmaschinen sind allerdings nur die Starrkörperbewegungen und die elastischen Verformungen relevant, da üblicherweise keine Versagensfälle, wie sie zum Beispiel bei Plastizität bei einem Bruch auftreten, betrachtet werden.

Bei der Betrachtung von Werkzeugmaschinen geht man von Systemen gelenkig verbundener fester Körper aus, wobei im Allgemeinen weder die Körper als ideal starr noch die Gelenke als ideal kinematisch angesehen werden und die Führungsbewegungen der Baugruppen von elastischen Verformungen überlagert werden.

Bei den Gestellkörpern von Werkzeugmaschinen handelt es sich um elastische Kontinua. Eine Beschreibung der elastischen Verformungen nach den Grundlagen der Kontinuumsmechanik führt zu nichtlinearen partiellen Differenzialgleichungen zweiter Ordnung, die nur für einfache Strukturen effizient berechenbare Lösungen besitzen. Für die Durchführung von effizienten numerischen Simulationen werden deshalb diskrete Formulierungen benötigt. Aus diesem Grund kommen in der Festkörpermechanik verschiedene Näherungsverfahren zum Einsatz. Diese liefern, ausgehend vom Konstruktionsdatensatz, Bewegungsgleichungen in Form von gewöhnlichen Differenzialgleichungen erster Ordnung. Im wesentlichen kommen drei Verfahrensarten zum Einsatz, die in Abbildung 7.12 dargestellt sind. Bei diesen Verfahren handelt es sich um Starrkörperansätze, um die Finite-Elemente-Methode (FEM) und um Kombinationen dieser zwei Verfahren den sogenannten elastischen Mehrkörpersystemen (MKS). Diese Verfahren unterscheiden sich durch die Ordnung und der Art der erzeugten Gleichungssysteme. Grundsätzlich stehen sich Verfahren, die nur kleine lineare Bewegungen berücksichtigen, Verfahren gegenüber, die auch große nichtlineare Bewegungen berücksichtigen. Diese Einflußfaktoren bestimmen die numerische Qualität und Effizienz der erzeugten Simulationsmodelle, wobei einer durch hohe Systemordnung und der Berücksichtigung von Nichtlinearitäten erreichten hohen Abbildungsgenauigkeit entsprechend lange Berechnungszeiten gegenüberstehen.

7.3.1.1 Starrkörpersysteme

Die Betrachtung einer mechanischen Struktur als System starrer Körper oder Massen führt direkt zu einer diskreten Darstellung der Bewegungsgleichungen.

Zur Diskretisierung von schwingungsfähigen Systemen mit wenigen starren Massen oder Trägheitsmomenten werden meist einfache Starrkörperansätze verwendet. Die Bewegungen dieser sogenannten Mehrmassensysteme werden durch einen entkoppelten und reduzierten Satz der Körperfreiheitsgrade und unter Vernachlässigung der Relativkinetik dargestellt. Mehrmassensysteme sind einfach zu beschreiben und werden häufig zur Berechnung von Torsionsschwingungen in rotatorischen Antriebssystemen verwendet, siehe [112, 115].

Die umfassendere Modellierung als Mehrkörpersystem (MKS) idealisiert das Festkörpersystem

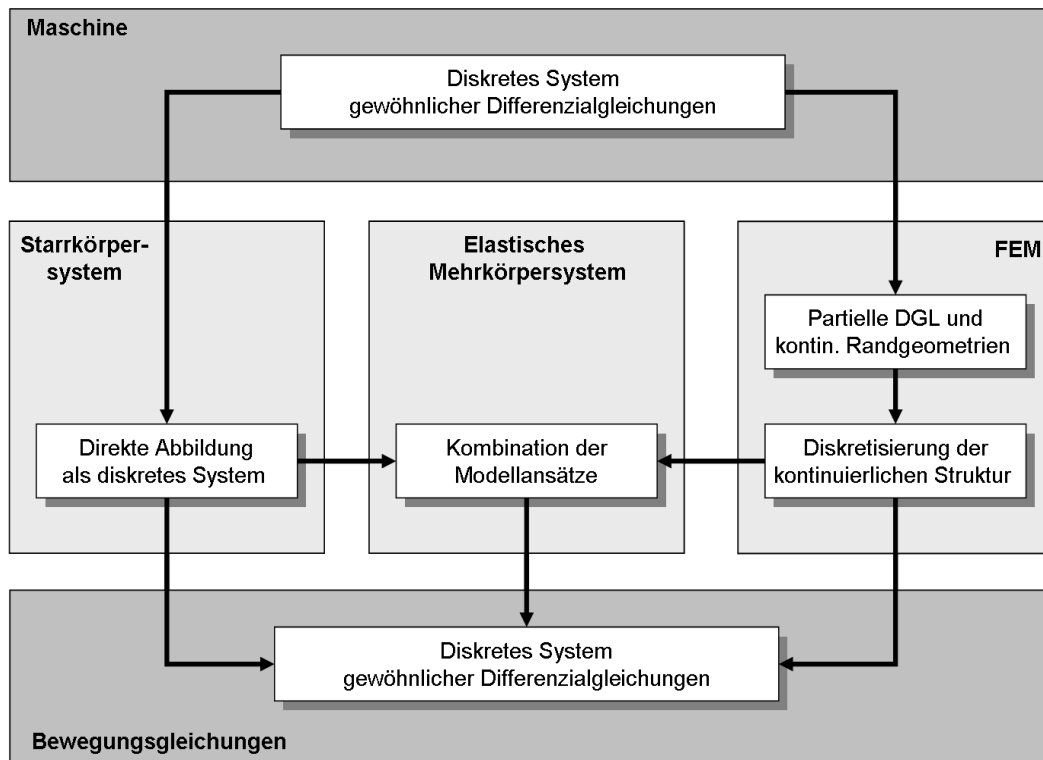


Abbildung 7.12: Modellierungsverfahren einer mechanischen Struktur nach [138]

durch starre Körper mit tensorischen Trägheitseigenschaften [184]. Bindungen zwischen den Körpern werden als masselos angenommen und greifen an diskreten Koppelpunkten der Körper an, wobei neben ideal-kinematischen Gelenken auch elastische und dämpfende Bindungen modelliert werden können. Die Angabe der Körperschwerpunkte, Trägheitssensoren und Bindungsknoten erfolgt in körperfesten Koordinatensystemen. Die räumlichen Bewegungen der Körper werden durch absolute Koordinaten direkt im Inertialsystem oder durch Relativkoordinaten zwischen den gelenkig verbundenen Körpern beschrieben.

Die mathematische Beschreibung der Bewegungsgleichungen eines MKS erfolgt unter Berücksichtigung der Relativkinematik mit den Prinzipien der virtuellen Arbeit nach d'Alembert bzw. der virtuellen Leistung nach Jourdain oder dem Verfahren nach Lagrange über das Aufstellen von Energiebilanzen [145]. Die mit starren MKS erzeugten Bewegungsgleichungen bilden sowohl die geometrisch nichtlinearen Verschiebungen der Massen und Bindungsknoten als auch die linearen Verschiebungen der Körper infolge der elastischen Verformungen der Bindungen ab. Da starre Körper betrachtet werden, besitzen diese keine Verformungsfreiheitsgrade, was in der Regel in Modelle mit geringer Dimension resultiert. Aus diesem Grund ist die numerische Berechnung solcher Modelle sehr effizient. Die Modellierung einer Maschine als starres MKS ist nur für solche Maschinen eine geeignete Beschreibungsform, deren Baugruppen, zumindest in dem betrachteten Frequenzbereich, als ideal starr angenommen werden können.

7.3.1.2 Finite-Elemente-Methode

Für die Beschreibung elastischer Strukturen liefert die Kontinuumsmechanik Ansätze in Form von partiellen Differenzialgleichungen. Diese Gleichungen sind allerdings nur lösbar, wenn die Ränder und Randbedingungen der Struktur analytisch beschreibbar sind, was nur für einfache Randgeometrien möglich ist. Um kontinuierliche Verformungsansätze auf beliebig komplexe Strukturen zu übertragen, wurde die Finite-Elemente-Methode (FEM) entwickelt. Das Vorgehen der FEM besteht in der Unterteilung einer kontinuierlichen Struktur in eine hohe Anzahl von finiten Elementen (FE) mit regelmäßiger Geometrie. Mit dieser Methode wird die kontinuierliche Struktur in eine raumdiskrete Darstellung überführt. Die finiten Elemente sind an diskreten Knotenpunkten mit ihren jeweiligen Nachbarelementen verbunden. Die Bewegungen der Struktur werden durch die Freiheitsgrade der FE-Knoten repräsentiert. Die Verbindung dieser Punkte bildet das sogenannte FE-Netz. Die Anzahl der verwendeten finiten Elemente bestimmt den Diskretisierungsgrad der Struktur. Ein hoher Diskretisierungsgrad geht mit einer Erhöhung der Qualität der Näherungslösung einher, zugleich wird aber auch die Anzahl der Freiheitsgrade und damit die Größe des entstehenden Gleichungssystems bestimmt.

Als Elementtypen, die zur Approximation der Struktur verwendet werden, existieren verschiedene Typen. Diese unterscheiden sich insbesondere durch die räumliche Dimension (0D - 3D), die Anzahl der Knoten, die Knotenfreiheitsgrade und der Art der Elementansatzfunktion, die das kontinuierliche Verschiebungsfeld im Inneren des Elementes in Abhängigkeit von den diskreten Knotenfreiheitsgraden beschreibt. Die Elementansatzfunktion sind oft linearer oder polynomieller Ordnung. Die verwendeten Elementtypen und Werkstoffparameter definieren die Massen- und Steifigkeitseigenschaften der FEM-Struktur, die im folgenden auch Strukturkörper oder mechanisches Strukturmodell genannt wird.

Mit den Materialparametern (Schubmodul, Querkontraktionszahl, Dichte) und den Knotenkoordinaten liefern die Verschiebungssätze nach dem Prinzip der virtuellen Arbeit die Steifigkeits-, Dämpfungs- und Massenmatrizen der Elemente, aus welchen anschließend die Koeffizientenmatrizen für die Bewegungsgleichungen der Gesamtstruktur erzeugt werden, die als gewöhnliches Differenzialgleichungssystem vorliegen.

Die FEM stellt heute im gesamten Maschinenbau ein Standardverfahren für strukturmechanische Berechnungen dar. Sie wird bei Werkzeugmaschinen erfolgreich zur Auslegung und Optimierung der Gestellstruktur eingesetzt.

7.3.1.3 Elastische Mehrkörpersysteme

Zur Simulation von Mehrkörpersystemen mit nichtlinearen Bewegungen und elastischen Körpern existieren verschiedene Ansätze. Die meisten Ansätze basieren auf der Synthese von geometrisch nichtlinearen Referenzbewegungen körperfester Bezugssysteme und linear-elastischen Verformungen der Körper. Dies führt zu MKS und FEM kombinierten Modellen. Die elastischen Körper werden dabei als FEM-Substrukturen modelliert und entweder direkt oder über einen Rayleigh-Ritz-Ansatz in das MKS eingebunden.

Bei der direkten Einbindung von FEM-Substrukturen werden die Knotenverschiebungen in kar-

tesischen Koordinaten relativ zum bewegten Bezugssystem angegeben. Die mit der FEM erzeugte Massen-, Steifigkeits- und Dämpfungsmatrix der elastischen Körper werden auf diese Weise direkt in die Bewegungsgleichungen der MKS integriert. Bei diesem Vorgehen wird die hohe Anzahl an Freiheitsgraden der FEM auf die MKS übertragen. Dies führt in Verbindung mit der nichtlinearen Darstellung der MKS jedoch zu sehr hohen Rechenzeiten, was die Anwendung dieser Methode nur auf Systeme mit einer kleinen Anzahl an Körpern praktikabel erscheinen lässt.

Ein effizienterer Ansatz liefert die Linearisierung der elastischen Verformungen entsprechend dem Rayleigh-Ritz-Ansatz. Dabei werden verallgemeinerte Koordinaten eingeführt, welche die elastischen Verformungen durch ortsabhängige Ansatzfunktionen beschreiben. Im Allgemeinen genügt eine im Vergleich zu den kartesischen Koordinaten der FEM-Struktur sehr kleine Anzahl verallgemeinerter Koordinaten, um die elastischen Verformungen zu approximieren, was zu numerisch erheblich effizienteren Darstellungsformen führt.

7.3.2 FEM-Modellierung des mechanischen Systems

Das elastische MKS stellt die umfassendste und vollständigste Beschreibungsform für das dynamische Verhalten von Werkzeugmaschinen dar. Allerdings ist die Erzeugung der Simulationsmodelle im Vergleich mit der FEM erheblich aufwendiger, da zusätzlich zur Modellierung der FEM-Substrukturen der Maschine weitere Arbeitsschritte bis zur Erzeugung eines MKS-Modells vonnöten sind.

Bei regelungstechnischen Analysen werden zudem im Allgemeinen nur lineare Bewegungen um einen Betriebspunkt betrachtet, da bei einer funktionierenden und im linearen Bereich arbeitenden Regelung die Regeldifferenzen klein sind. Für eine Regelsimulation genügt daher die Berechnung der im Vergleich zu den Führungsbewegungen kleinen Regeldifferenzen mit linear-elastischen Modellen. Aus diesem Grund ist die Verwendung der FEM zur Simulation des Strukturverhaltens in Verbindung mit einer Regelung bereits in verschiedenen Arbeiten gegenüber anderen Ansätzen bevorzugt worden, siehe dazu [26, 27, 56, 187] und wird auch in Hinblick auf die industrielle Anwendung als geeigneteres Verfahren erachtet [83].

Bei der FEM-Darstellung einer Werkzeugmaschine werden die mit Volumen- oder Schalenelementen vernetzten Gestellbausteine, wie zum Beispiel das Maschinenbett oder der Vorschubschlitten, Strukturkörper genannt. Aber auch die mit eindimensionalen Balkenelementen vernetzten Bausteine, wie die Übertragungswellen der Antriebsstränge einer Gewindespindel werden mit diesem Begriff bezeichnet. Innerhalb einer Baugruppe einer Maschine sind die Strukturkörper durch Verbindungselemente mit spezifischen Gelenkeigenschaften miteinander gekoppelt. Diese Maschinenelemente werden im FEM-Modell auf ihre elastischen Bindungseigenschaften abstrahiert und durch Federelemente abgebildet.

Der Erstellungsaufwand nimmt mit der konstruktiven Komplexität der einzelnen Strukturkörper und der Anzahl der Verbindungselemente zu. Dabei bietet der heutige Stand der Technik die Möglichkeit, die Strukturkörper CAD-gestützt zu modellieren und mit Hilfe von Netzgeneratoren automatisiert zu diskretisieren. Dem gegenüber erfordert die Modellierung der Verbindungselemente zur Erzeugung von Baugruppenmodellen üblicherweise umfangreiche Benutzereingaben

im FEM-Präprozessor. Dieser Aufwand kommt zustande, da die Kontaktknoten an den Strukturkörpern ausgewählt, die Federelemente definiert und diesen Steifigkeitswerte zugewiesen werden müssen. Bei jeder Modifikation des Strukturkörpers müssen die Verbindungselemente wieder gelöst und neu erzeugt werden.

Um den manuellen Aufwand bei der Modellierung von Gestellbaugruppen zu reduzieren, wurden in [158] Modellbausteine entwickelt, mit deren Hilfe Kontaktflächen und Stetigkeitsparameter für Verbindungselemente mit einem Minimum an Benutzerinteraktion definiert und die Federverbindung automatisiert erzeugt werden kann. Dazu ist eine eindeutige netzunabhängige Identifikation der Kontaktflächen zwischen den Komponenten erforderlich. Ein Lösungsansatz hierfür liefert der in [158] eingeführte Ansatz der netzunabhängigen Kopplung.

7.3.2.1 Netzunabhängige Kopplung

Verbindungselemente werden im FEM-Modell in abstrahierter Form durch Federelemente modelliert. Dabei wird ihre Gesamtsteifigkeit auf die diskreten Federsteifigkeiten konzentriert. Da die Spannungskonzentration bzw. die lokale Unstetigkeit in der Steifigkeitsmatrix an den Verbindungsknoten zu schwer abschätzbaren numerischen Fehlern führen, empfiehlt es sich, eine Feder nicht direkt an einen einzelnen Knotenpunkt einer FEM-Struktur anzukoppeln. Numerisch und vernetzungstechnisch praktikabler ist eine netzunabhängige Anbindung des Federelements über ein Starrkörperelement (RBE²), wie in Abbildung 7.13 dargestellt. Ein RBE stellt dabei eine besondere Form von inneren Zwangsbedingungen (MPCs³) dar.

Die MPCs stellen kinematische Abhängigkeiten zwischen mehreren Freiheitsgraden her. Dazu werden die betrachteten Freiheitsgrade in unabhängige und abhängige Freiheitsgrade unterteilt. Die MPCs können dann in der Form eines Skalarproduktes angegeben werden:

$$\begin{bmatrix} \mathbf{a}_{UN} \\ \mathbf{a}_{AB} \end{bmatrix}^T \cdot \begin{bmatrix} \mathbf{u}_{UN} \\ \mathbf{u}_{AB} \end{bmatrix} = 0. \quad (7.22)$$

Dabei sind \mathbf{u}_{UN} und \mathbf{u}_{AB} die Verschiebungen der unabhängigen bzw. der abhängigen Freiheitsgrade mit den Linearkoeffizienten \mathbf{a}_{UN} bzw. \mathbf{a}_{AB} .

Für ein RBE errechnen sich die Linearkoeffizienten aus den Knotenkoordinaten in dem Sinne, dass sich die Knoten des RBEs relativ zueinander nicht bewegen, sich also wie Punkte eines starren Körpers verhalten. Die sechs Starrkörperfreiheitsgrade (Referenzfreiheitsgrade) des RBE werden einem strukturfremden Knoten, dem sogenannten Referenzknoten, zugeordnet, der, wie in Abbildung 7.13 dargestellt, als Koppelknoten für die Feder dient.

Je nachdem, ob der Koordinatensatz des Referenzknotens gemäß Gleichung (7.22) den unabhängigen oder den abhängigen Freiheitsgraden (\mathbf{u}_{UN} bzw. \mathbf{u}_{AB}) entspricht, sind zwei Typen von RBEs zu unterscheiden. Ein RBE vom Typ 1 besitzt einen unabhängigen Referenzknoten und gibt somit die Verschiebungen der verbundenen (abhängigen) Strukturknoten explizit vor. Die Kontaktfläche des Strukturkörpers kann sich deshalb nicht mehr elastisch verformen.

²In Englisch: Rigid Body Element

³Englisch: Multipoint Constraints

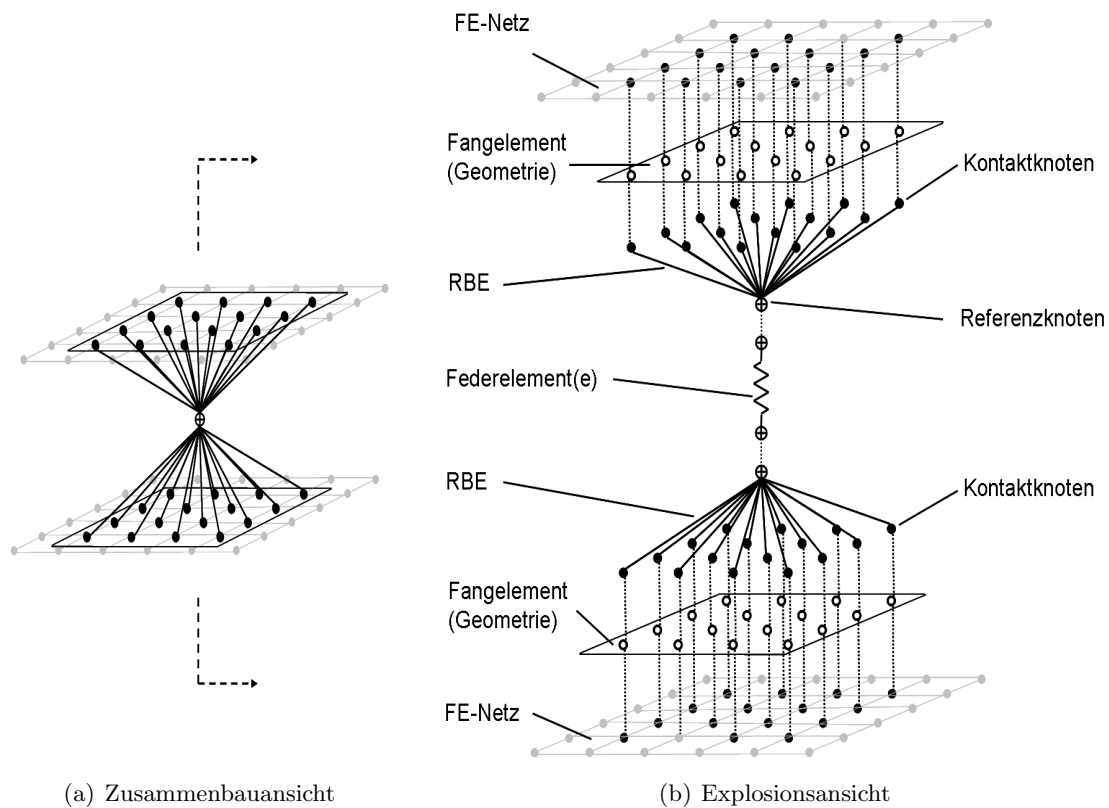


Abbildung 7.13: Netzunabhängige Kopplung von FE-Strukturen nach [138]

Bei einem RBE vom Typ 2 ist der Referenzknoten abhängig von den (unabhängigen) Strukturknoten der Kontaktfläche. Die Kontaktfläche des Strukturkörpers behält in diesem Fall die durch das FEM-Netz gegebene Elastizität, während der Referenzknoten eine aus den unabhängigen Knotenverschiebungen interpolierte Verschiebung erfährt. Diese Knotenverschiebung ist im Allgemeinen etwas größer als im Fall eines RBE vom Typ 1. Abbildung 7.14 stellt die beiden Typen von RBEs graphisch dar.

Zur Erzeugung einer RBE-Verbindung im FEM-Modell müssen die Kontaktknoten am Strukturkörper identifiziert werden. In [158] wird dafür ein Kopplungsautomatismus entwickelt. Dabei werden die Kontaktknoten nicht explizit ausgewählt, sondern die Kontaktfläche durch ein geometrisches Flächenelement gekennzeichnet, das als Fangelement dient, zur Veranschaulichung siehe Abbildung 7.13. Ein Algorithmus wählt Strukturknoten innerhalb des Flächenelementes als Kontaktknoten aus. Sind die Kontaktflächen eines Verbindungselementes durch Flächenelemente gekennzeichnet und die Federsteifigkeit definiert, können die Referenzknoten, die RBEs und die Federamente, d.h. ein gesamtes Verbindungselement, automatisch erzeugt werden.

Die einzige Schnittstelleninformation einer netzunabhängigen Kopplung dieser Art ist das Fangelement an der Kontaktfläche des Strukturkörpers. Da die Kontaktknoten eindeutig mit den Fangelementen assoziierbar sind, können die Fügepartner durch Löschen des RBEs jederzeit getrennt, beliebig modifiziert und anschließend der Kontakt automatisch wiederhergestellt werden.

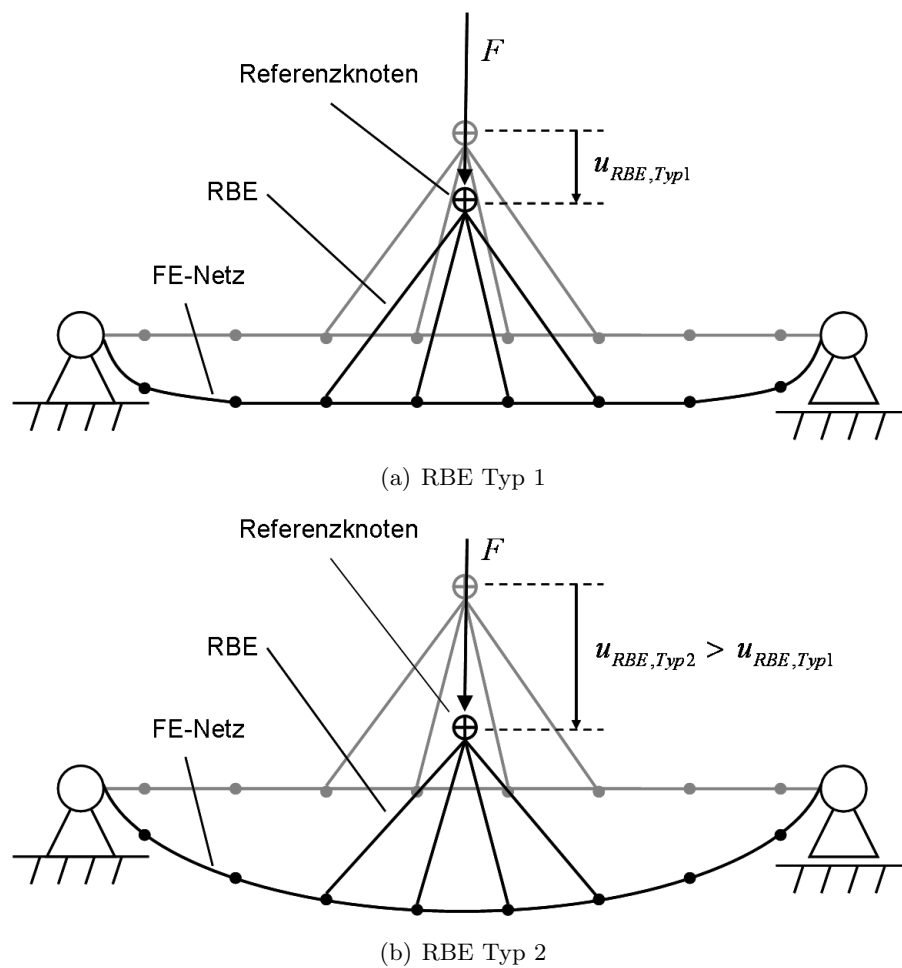


Abbildung 7.14: Explizite und interpolierende Starrkörperelemente (Typ 1 bzw. Typ 2) nach [158]

7.3.2.2 Parametrische FEM-Modellbausteine

Die Struktorkörper in Antriebssträngen sind vornehmlich Wellen (Motorwelle, Getriebewellen, Gewindespindel u.a.), die durch eindimensionale Netze mit Balkenelementen dargestellt werden. Im Gegensatz zu Strukturen, die durch Volumen- und Schalenelementen dargestellt werden, existieren keine automatischen Diskretisierungsverfahren. Die Wellenstrukturen müssen deshalb explizit im FEM-Präprozessor modelliert und unter Verwendung von Querschnittsbibliotheken und -editoren parametrisiert werden. Darüber hinaus beinhalten Antriebsstrukturen zusätzliche Verbindungselemente unterschiedlicher Komplexität (Lager, Kupplung, Getriebestufe u.a.).

Da die Erzeugung der Wellenstrukturen weitgehend manuell erfolgt, ist zur Erweiterung bereits vorhandener FEM-Modelle von Gestellbaugruppen um detaillierte Antriebsstrukturen ein erheblicher zusätzlicher Zeitaufwand erforderlich. Für die Modellierung von Werkzeugmaschinen mit hohem Detaillierungsgrad sind deshalb effiziente Modellierungswerkzeuge notwendig. Auf diese Vorgehensweise aufbauend wurden in [138] problemorientierte Modellbausteine zum Erzeugen und Editieren von FEM-Strukturen für Werkzeugmaschinen, insbesondere der Antriebsstränge, entwickelt. Tabelle 7.1 gibt eine Übersicht über die wichtigsten Komponentenklassen für Gestell-

und Antriebsstrukturen und die entsprechenden Diskretisierungsansätze für deren Modellbausteine. Eine ausführliche Beschreibung der verschiedenen Komponentenklassen ist in [138] zu finden. Die in [158, 138] vorgestellten Modellbausteine gehen über das reine Erzeugen von FEM-

Tabelle 7.1: Komponentenklassen für Gestell und Antriebsstrukturen sowie die entsprechenden Diskretisierungsansätze für deren Modellbausteine nach [138]

Komponentenklasse	Diskretisierungsansatz
3D-Strukturkörper (z.B. Maschinenbett, Vorschubschlitten)	Strukturkörper mit frei vernetzter 3D-Geometrie (Volumen und Schaltenelementen)
Allgemeines Verbindungselement	Verbindungselement mit frei belegbaren elastischen Bindungen in allen sechs Raumrichtungen
Welle (Motorwelle, Vorschubspindel u.a.)	Strukturkörper aus Balkenelementen
Lager	Verbindungselement mit elastischen Bindungen in den Radialrichtungen und ggf. in axialer Richtung
Kupplung	Verbindungselement mit elastischer Bindung im Torsionsfreiheitsgrad
Linearführung	Verbindungselement mit elastischen Bindungen in lateraler und vertikaler Richtung
Kugelgewindetrieb	Verbindungselement mit Schraubfreiheitsgrad und elastischen Bindungen in den sechs Raumfreiheitsgraden
Getriebestufe (Zahnriemen, Zahnrad)	Verbindungselement mit kinematischer und elastischer Kopplung der An- und Abtriebsknoten

Strukturen hinaus. Da mit diesen Modellbausteinen auch deren nachträgliche Veränderbarkeit anhand der Eingabeparameter realisiert werden kann, wird der Aufbau parametrischer FEM-Modelle ermöglicht. Die FEM-Netze der einzelnen Komponenten können damit effizient editiert werden, ohne dass die jeweiligen Teilstrukturen durch manuelles Löschen von Kopplungen aus den Baugruppenmodell herausgetrennt werden müssen. Diese Änderungsprozesse lassen sich durch Modellbausteine mit parametrischer Funktionalität automatisieren. Dazu muss vorausgesetzt werden können, dass die Modellbausteine nach der Aktualisierung der FEM-Netze die zuvor definierten Kopplungen automatisch wiederherstellen können, wozu eine eindeutige netzunabhängige Identifikation der Kontaktflächen zwischen den Komponenten erforderlich ist.

7.3.2.3 Datenmodell für parametrische FEM-Modellbausteine

Die Grundlage parametrischer FEM-Modelle bildet ein objektorientiertes Datenmodell, das die Datenstruktur von FEM-Strukturmodellen beschreibt. Dazu werden für die Komponenten der Maschine (Strukturkörper und die Verbindungselemente) Modellbausteine entwickelt, welche die klassenspezifischen FEM-Strukturmodelle anhand vorgegebener Eingabeparameter erzeugen und mit Hilfe der netzunabhängigen Kopplung in die Baugruppe einfügen.

Zur Beschreibung der Modellbausteine werden objektorientierte Datenmodelle verwendet, die mit Hilfe der graphischen Modellierungssprache UML⁴ dargestellt werden, siehe [139]. Dabei

⁴Englisch: Unified Modelling Language

handelt es sich bei der UML um einen Sprachstandard zur Modellierung von Systemen und Prozessen, der auch bei komplexen Datenstrukturen anschauliche und widerspruchsfreie Darstellungen ermöglicht.

Ein weiterer Vorteil dieser Sprache liegt in der Neutralität in Bezug auf die Syntax spezifischer Programmiersprachen, was eine Übertragbarkeit der entwickelten Datenmodelle auf jedes Softwaresystem das über eine Programmierschnittstelle (API)⁵ verfügt, garantiert. In Abbildung 7.15 ist die Anwendung der Modellbausteine in Verbindung mit einem konventionellen FEM-Programmsystem dargestellt. Mit Hilfe des API wird in diesem Fall ein anwendungsspe-

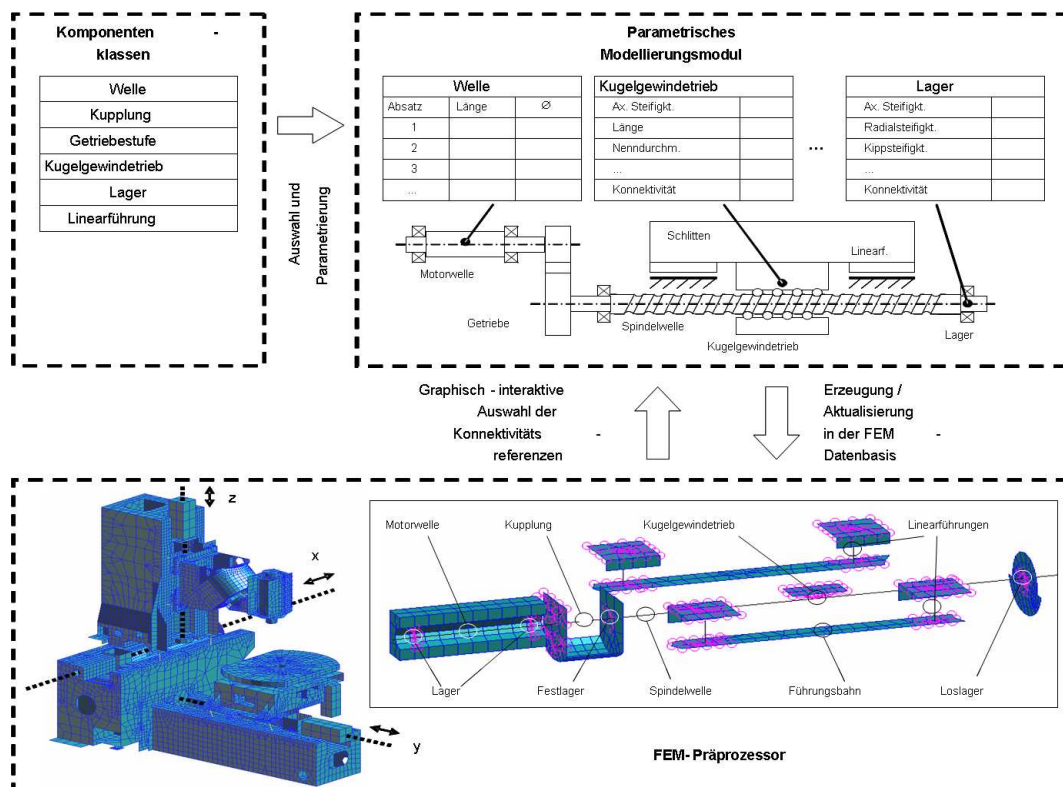


Abbildung 7.15: Arbeit mit Modellbausteinen in Verbindung mit konventionellen FEM-Programmsystemen nach [138]

zifisches Hilfsprogramm (Parametrischer Modelleditor) zur Kommunikation mit dem Benutzer und der FEM-Datenbasis verwendet. Die API dient dazu, die erforderlichen Werkzeuge zur Programmierung von graphischen Benutzeroberflächen und zur Manipulation des FEM-Modells zur Verfügung zu stellen. Mit dem parametrischen Modelleditor werden die Modellbausteine benutzeraktiv ausgewählt, parametrisiert und instantiiert. Dabei werden die Konnektivitätsreferenzen (die Kontaktstellen der angrenzenden Strukturen) für den zu erzeugenden Modellbaustein graphisch-interaktiv im FEM-Präprozessor selektiert. Beim Initiieren erzeugt der parametrische Modelleditor eine klassenspezifische FEM-Struktur, koppelt diese mit dem im FEM-Präprozessor bereits vorhandenen FEM-Modell und speichert alle Erzeugungsinformationen, so dass die Modellobjekte zu jedem späteren Zeitpunkt wieder verändert werden können. Die Realisierung

⁵Englisch: Application Programming Interface

des parametrischen Modelleditors orientiert sich an den spezifischen Gegebenheiten des jeweils verwendeten Simulationssystems sowie dessen API.

7.3.3 Bewegungsgleichungen im Zustandsraum

Zur Modellierung des mechanischen Systems wird ein FEM-Modell mit den in den vorhergehenden Abschnitten beschriebenen Eigenschaften verwendet. Somit liegt ein System zweiter Ordnung der Form (4.1) mit Massen- und Steifigkeitsmatrix \mathbf{M} bzw. \mathbf{K} vor, das alle unabhängigen und ansprechbaren Freiheitsgrade beinhaltet.

Bei der Modellierung von Werkzeugmaschinen werden Schalenelemente und zur Verbindung von Maschinenteilen Federelemente verwendet. Da die Federelemente nicht massebehaftet sind und bei den Schalenelementen die Masse nur auf die translatorischen Freiheitsgrade verteilt werden, ist die Massenmatrix \mathbf{M} im Allgemeinen singulär, so dass das System unter der Annahme eines linearen Verhaltens der Störbewegungen zunächst als ein zwangserregtes, lineares, zeitinvariantes und gewöhnliches Algebro-Differenzialgleichungssystem zweiter Ordnung vorliegt.

Bei der Modellierung der Werkzeugmaschinenstruktur mit finiten Elementen wird zunächst auf die Modellierung der Dämpfung verzichtet. Diese kann in Form einer Dämpfungsmatrix \mathbf{D} hinzugefügt werden, siehe dazu Abschnitt 4.3. Weiterhin werden, im Hinblick auf die Kopplung des Gleichungssystems mit dem des regelungs- und elektrotechnischen Gleichungssystems, die seitens der Antriebe anliegenden Motormomente \mathbf{M}_M und äußere Störkräfte \mathbf{F}_P , die auf einen Punkt P der Maschine wirken, über die Eingangsvektoren \mathbf{u}_A und \mathbf{u}_S getrennt eingeleitet, wobei

$$\mathbf{u}_A := [\mathbf{M}_M] \quad \text{und} \quad \mathbf{u}_S := [\mathbf{F}_P] \quad (7.23)$$

gilt. Dabei ist es üblich, den Tool Center Point (TCP) der Maschine als Punkt P zu betrachten. Analog werden am Ausgang des mechanischen Systems die an die Regelung zurückgeführten Messgrößen der Lagen \mathbf{x}_i und Drehzahlen \mathbf{n}_i und die Bewegungen \mathbf{x}_P am Punkt P in zwei separate Ausgangsvektoren \mathbf{y}_M und \mathbf{y}_S ausgeleitet:

$$\mathbf{y}_M := \begin{bmatrix} \mathbf{x}_i \\ \mathbf{n}_i \end{bmatrix} \quad \text{und} \quad \mathbf{y}_S := [\mathbf{x}_P]. \quad (7.24)$$

Mit den Abgriffsmatrizen $\mathbf{C}_{p,M}$, $\mathbf{C}_{v,M}$ (Messgrößenausgänge) und $\mathbf{C}_{p,S}$, $\mathbf{C}_{v,S}$ (Störstellenausgänge) zur Ausleitung der Messgrößen bzw. der Bewegungen an den Störstellen stellt sich die Ausgangsgleichung des Systems mit zwei Partitionen dar:

$$\begin{bmatrix} \mathbf{y}_M \\ \mathbf{y}_S \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{p,M} \\ \mathbf{C}_{p,S} \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{C}_{v,M} \\ \mathbf{C}_{v,S} \end{bmatrix} \dot{\mathbf{x}}. \quad (7.25)$$

Mit den Eingriffsmatrizen \mathbf{F}_A und \mathbf{F}_S zur formalen Aufspaltung der Antriebs- bzw. der Störkräfte auf die betreffenden mechanischen Knotenfreiheitsgrade ergibt sich die Beschreibung des gedämpften Gesamtsystems zu einem Differenzialgleichungssystem zweiter Ordnung

der Form:

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{x}} + \mathbf{D}\dot{\mathbf{x}} + \mathbf{K}\mathbf{x} &= [\mathbf{F}_A \ \mathbf{F}_S] \begin{bmatrix} \mathbf{u}_A \\ \mathbf{u}_S \end{bmatrix} \\ \mathbf{y} = \begin{bmatrix} \mathbf{y}_M \\ \mathbf{y}_S \end{bmatrix} &= \begin{bmatrix} \mathbf{C}_{p,M} \\ \mathbf{C}_{p,S} \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{C}_{v,M} \\ \mathbf{C}_{v,S} \end{bmatrix} \dot{\mathbf{x}}. \end{aligned} \quad (7.26)$$

Das erhaltene System beschreibt die Maschinenstruktur inklusive aller modellierter Vorschubachsen. Durch Ansprechen der entsprechenden rotatorischen Freiheitsgrade der Antriebe über die Eingangsvektoren \mathbf{u}_A und \mathbf{u}_S können nun Motormomente bzw. äußere Störkräfte auf die Struktur bzw. auf die gewünschte Vorschubachse gegeben werden, um somit beispielsweise eine Analyse bezüglich der Positioniergenauigkeit und der Regelkreisdynamik durchführen zu können. Für die Einbindung der Bewegungsgleichungen des mechanischen Systems in das mechatronische Gesamtmodell und die gemeinsame Lösung des gekoppelten Gleichungssystems wird die Darstellung der Gleichungen im Zustandsraum verwendet. Hierzu werden für die Lagen \mathbf{x} und die Geschwindigkeiten $\dot{\mathbf{x}}$ die Zustandsvariablen \mathbf{z}_1 und \mathbf{z}_2 eingeführt und das Differenzialgleichungssystem zweiter Ordnung in ein System erster Ordnung überführt. Dazu wird der Zustandsvektor \mathbf{z} als

$$\mathbf{z} = \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{x} \\ \dot{\mathbf{x}} \end{bmatrix} \quad (7.27)$$

definiert. Mit diesem Zustandsvektor lässt sich das Differenzialgleichungssystem zweiter Ordnung in ein Differenzialgleichungssystem erster Ordnung der Form

$$\begin{aligned} \mathbf{E}\dot{\mathbf{z}} &= \mathbf{A}\mathbf{z} + \mathbf{B}\mathbf{u} \\ \mathbf{y} &= \mathbf{C}\mathbf{z} \end{aligned} \quad (7.28)$$

transformieren, wobei

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} \mathbf{0} & \mathbf{I}_n \\ -\mathbf{K} & -\mathbf{D} \end{bmatrix}, & \mathbf{B} &= [\mathbf{B}_A \ \mathbf{B}_S] = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{F}_A & \mathbf{F}_S \end{bmatrix} \\ \mathbf{E} &= \begin{bmatrix} \mathbf{I}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{M} \end{bmatrix}, & \mathbf{C} &= \begin{bmatrix} \mathbf{C}_M \\ \mathbf{C}_S \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{p,M} & \mathbf{C}_{v,M} \\ \mathbf{C}_{p,S} & \mathbf{C}_{v,S} \end{bmatrix} \end{aligned} \quad (7.29)$$

gilt. Für den Eingang und den Ausgang gilt dabei die Partitionierung:

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_A \\ \mathbf{u}_S \end{bmatrix} \quad \text{bzw.} \quad \mathbf{y} = \begin{bmatrix} \mathbf{y}_M \\ \mathbf{y}_S \end{bmatrix}. \quad (7.30)$$

Dabei ist \mathbf{I}_n die $n \times n$ -Einheitsmatrix, $\mathbf{0}$ eine Nullmatrix jeweils geeigneter Dimension, und \mathbf{D} im ungedämpften Fall gleich Null, siehe dazu auch Abschnitt 3.1, in dem auf die Transformation von Systemen zweiter Ordnung in Systeme erster Ordnung näher eingegangen wird. Ist die Massenmatrix \mathbf{M} nicht singulär, lässt sich das System (7.28) auch schreiben als

$$\begin{aligned} \dot{\mathbf{z}} &= \check{\mathbf{A}}\mathbf{z} + \check{\mathbf{B}}\mathbf{u}, \\ \mathbf{y} &= \mathbf{C}\mathbf{z}, \end{aligned} \quad (7.31)$$

wobei

$$\begin{aligned}\check{\mathbf{A}} &= \begin{bmatrix} \mathbf{0} & \mathbf{I}_n \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{D} \end{bmatrix} \\ \check{\mathbf{B}} = [\check{\mathbf{B}}_A \check{\mathbf{B}}_S] &= \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{M}^{-1}\mathbf{F}_A & \mathbf{M}^{-1}\mathbf{F}_S \end{bmatrix}.\end{aligned}\quad (7.32)$$

In Abbildung 7.16 ist das Blockschaltbild der in Zustandsdarstellung transformierten Bewegungsgleichungen (7.31) dargestellt. Mit Hilfe dieser Darstellung ist es nun möglich, das mechanische Modell mit dem regelungs- und elektrotechnischem Modell, wie es in Abbildung 7.6 dargestellt ist, beziehungsweise dessen Gleichungssystem (7.18) über den Antriebsstrang \mathbf{u}_A und den Messgrößenausgang \mathbf{y}_M zu koppeln. Der Störstellenein- (\mathbf{u}_S) und -ausgang (\mathbf{y}_S) kann zur

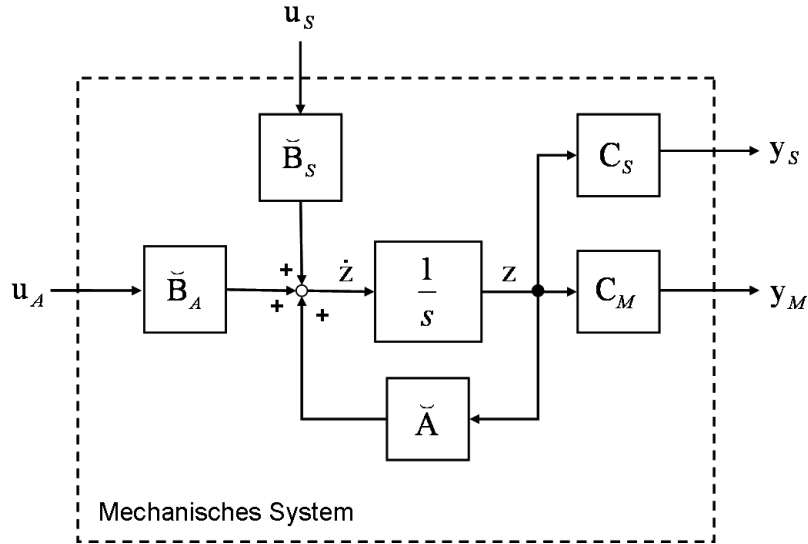


Abbildung 7.16: Blockschaltbild der Bewegungsgleichungen in Zustandsdarstellung nach [138]

Analyse des Störübertragungsverhaltens der Regelkreise oder der dynamischen Relativnachgiebigkeit zwischen Werkzeug und Werkstück verwendet werden. Über diese Schnittstelle können darüber hinaus auch nichtlineare Reibkraftmodelle mit Rückkopplungseffekten angeschlossen und damit zum Beispiel das Verhalten der Maschine unter betriebsähnlichen Lasten abgeschätzt werden.

7.4 Modellordnungsreduktion

Die Modellierung elastischer Strukturen mit der FEM führt zu sehr großen Gleichungssystemen, deren Lösung viel Rechenzeit in Anspruch nehmen. Diese Zeiten vervielfachen sich noch, wenn iterative Optimierungsrechnungen oder Simulationen im Zeitbereich durchgeführt werden. Dies stellt sich bei der Verwendung von FEM-Modellen zur Beschreibung elastischer Mehrkörpersysteme oder der Regelstrecken von Regelsystemen als problematisch dar. Ausgehend von diesem Sachverhalt wurden auf dem Gebiet der Mehrkörpermechanik und der Regelungstechnik Verfahren entwickelt, die große Modelle reduzieren. Dabei werden ausgehend vom gegebenen System

von Bewegungsgleichungen hoher Dimension ein Gleichungssystem bestimmt, dessen Systemmatrizen eine deutlich geringere Dimension aufweisen. Dadurch wird meist eine erheblich schnellere Lösung des Gleichungssystems erreicht. Bei einer Ordnungsreduktion sollten zudem wichtige Eigenschaften des Systems erhalten bleiben.

Ziel der Modellreduktion ist dabei, dass die Differenz der Ausgänge des originalen und des reduzierten Systems $\|\mathbf{y} - \hat{\mathbf{y}}\|$ für alle zulässigen Eingangssignale \mathbf{u} , bezüglich einer geeignet gewählten Norm $\|\cdot\|$, klein ist.

Bei der Reduktion eines Systems erster Ordnung der Form (7.28) wird ein System mit $r \ll 2n$ gesucht, das folgende Form aufweist:

$$\begin{aligned}\tilde{\mathbf{E}}\dot{\hat{\mathbf{z}}} &= \tilde{\mathbf{A}}\hat{\mathbf{z}} + \tilde{\mathbf{B}}\mathbf{u}, \\ \hat{\mathbf{y}} &= \hat{\mathbf{C}}\hat{\mathbf{z}},\end{aligned}\tag{7.33}$$

wobei die Matrizen $\tilde{\mathbf{E}}, \tilde{\mathbf{A}} \in \mathbb{R}^{r \times r}$ regulär sind und

$$\tilde{\mathbf{B}} = [\tilde{\mathbf{B}}_A \ \tilde{\mathbf{B}}_S] \in \mathbb{R}^{r \times m}, \quad \hat{\mathbf{C}} = \begin{bmatrix} \hat{\mathbf{C}}_M \\ \hat{\mathbf{C}}_S \end{bmatrix} \in \mathbb{R}^{q \times r}\tag{7.34}$$

gilt. Die reduzierten Systemmatrizen und der Zustand $\hat{\mathbf{z}}(t)$ hat im reduzierten System die Dimension r . Der Eingang \mathbf{u} bleibt unverändert und der Ausgang $\hat{\mathbf{y}} = [\hat{\mathbf{y}}_M \ \hat{\mathbf{y}}_S]^T$ stellt, wie oben gefordert, eine Approximation an den Ausgang des Originalsystems dar.

Ist die reduzierte Systemmatrix $\tilde{\mathbf{E}}$ regulär, lässt sich das System (7.33) auch schreiben als

$$\begin{aligned}\dot{\hat{\mathbf{z}}} &= \hat{\mathbf{A}}\hat{\mathbf{z}} + \hat{\mathbf{B}}\mathbf{u}, \\ \hat{\mathbf{y}} &= \hat{\mathbf{C}}\hat{\mathbf{z}},\end{aligned}\tag{7.35}$$

wobei

$$\hat{\mathbf{A}} = \tilde{\mathbf{E}}^{-1} \tilde{\mathbf{A}}\tag{7.36}$$

und

$$\hat{\mathbf{B}} = [\hat{\mathbf{B}}_A \ \hat{\mathbf{B}}_S] = [\tilde{\mathbf{E}}^{-1} \tilde{\mathbf{B}}_A \ \tilde{\mathbf{E}}^{-1} \tilde{\mathbf{B}}_S]\tag{7.37}$$

gilt.

Alternativ lässt sich ein reduziertes System zweiter Ordnung der Form (7.26) vor der Transformation in ein System erster Ordnung der Form (7.28) reduzieren. Bei der Reduktion eines Systems zweiter Ordnung wird ein System mit $r \ll n$ gesucht, das folgende Form aufweist:

$$\begin{aligned}\hat{\mathbf{M}}\ddot{\hat{\mathbf{x}}} + \hat{\mathbf{D}}\dot{\hat{\mathbf{x}}} + \hat{\mathbf{K}}\hat{\mathbf{x}} &= \hat{\mathbf{F}}\mathbf{u} \\ \hat{\mathbf{y}} &= \hat{\mathbf{C}}_p\hat{\mathbf{x}} + \hat{\mathbf{C}}_v\dot{\hat{\mathbf{x}}},\end{aligned}\tag{7.38}$$

wobei die Systemmatrizen $\hat{\mathbf{M}}, \hat{\mathbf{D}}, \hat{\mathbf{K}} \in \mathbb{R}^{r \times r}$ regulär sind und

$$\hat{\mathbf{F}} = [\hat{\mathbf{F}}_A \ \hat{\mathbf{F}}_S] \in \mathbb{R}^{r \times m}, \quad \hat{\mathbf{C}}_p = \begin{bmatrix} \hat{\mathbf{C}}_{p,M} \\ \hat{\mathbf{C}}_{p,S} \end{bmatrix} \in \mathbb{R}^{q \times r}, \quad \hat{\mathbf{C}}_v = \begin{bmatrix} \hat{\mathbf{C}}_{v,M} \\ \hat{\mathbf{C}}_{v,S} \end{bmatrix} \in \mathbb{R}^{q \times r}\tag{7.39}$$

gilt. Die reduzierten Systemmatrizen und der Zustand $\hat{\mathbf{x}}$ haben im reduzierten System die Dimension r . Der Eingang \mathbf{u} bleibt unverändert und der Ausgang $\hat{\mathbf{y}} = [\hat{\mathbf{y}}_M \ \hat{\mathbf{y}}_S]^T$ stellt, wie oben gefordert, eine Approximation an den Ausgang des Originalsystems dar.

Ist die reduzierte Massenmatrix \hat{M} regulär, lässt sich das reduzierte System zweiter Ordnung als reduziertes System erster Ordnung der Form (7.35) mit $\hat{z} \in \mathbb{R}^{2r}$ und einer reduzierten Dimension $2r$ schreiben, wobei

$$\hat{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_n \\ -\hat{M}^{-1}\hat{K} & -\hat{M}^{-1}\hat{D} \end{bmatrix}, \quad (7.40)$$

$$\hat{B} = [\hat{B}_A \ \hat{B}_S] = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \hat{M}^{-1}\hat{F}_A & \hat{M}^{-1}\hat{F}_S \end{bmatrix} \quad (7.41)$$

und

$$\hat{C} = \begin{bmatrix} \hat{C}_M \\ \hat{C}_S \end{bmatrix} = \begin{bmatrix} \hat{C}_{p,M} & \hat{C}_{v,M} \\ \hat{C}_{p,S} & \hat{C}_{v,S} \end{bmatrix}. \quad (7.42)$$

In Kapitel 5 werden die Grundlagen der Modellordnungsreduktion ausführlich dargestellt und in Kapitel 10 werden speziell für die Simulation von Werkzeugmaschinen neu entwickelte Modellordnungsreduktionsverfahren vorgestellt.

7.5 Kopplung der Gleichungssysteme

Für die Analyse des mechatronischen Gesamtsystems lassen sich das regelungs- und elektrotechnische (7.18) sowie das reduzierte mechanische Gleichungssystem (7.23) koppeln. In Abbildung 7.17 ist das Blockschaltbild der zusammengeführten Gleichungssysteme im Zustandsraum dargestellt. Die aus dem regelungs- und elektrotechnischen System mit Hilfe des Ausgangsvektors

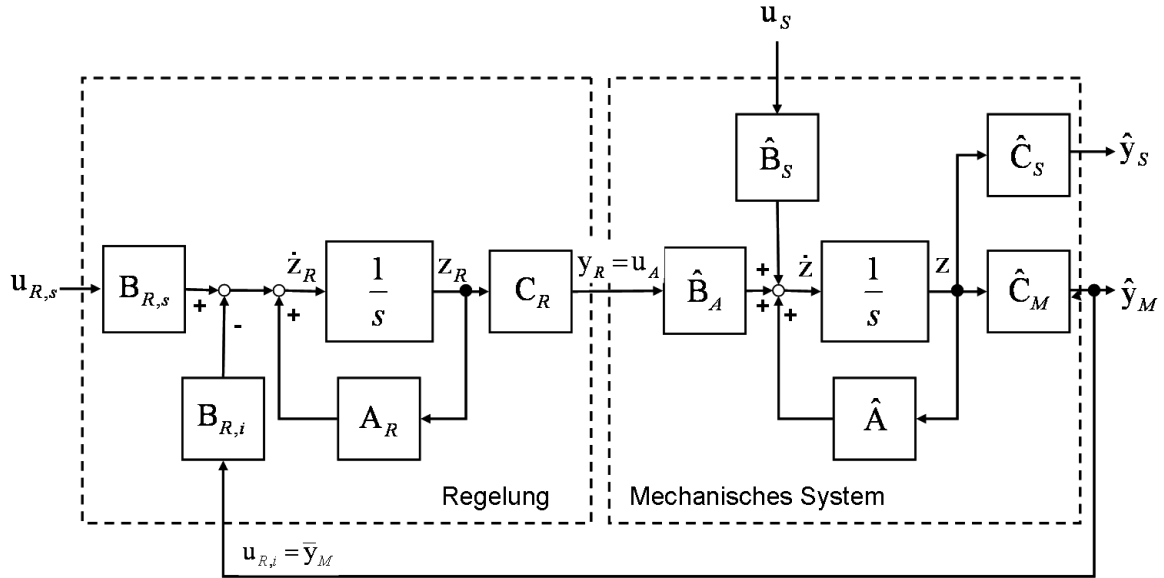


Abbildung 7.17: Blockschaltbild des regelungs- und elektrotechnischen sowie des reduzierten mechanischen Gleichungssystems im Zustandsraum nach [138]

y_R ausgeleiteten Motormomente M_M werden an die Eingänge des mechanischen Systems übertragen. Dann ist

$$u_A = y_R := [M_M]. \quad (7.43)$$

Die in den Ausgangsvektor $\hat{\mathbf{y}}_M$ des reduzierten mechanischen Systems ausgeleiteten approximierten Messgrößen der Lagen $\hat{\mathbf{x}}_i$ und der Drehzahlen $\hat{\mathbf{n}}_i$ werden an den Istgrößen-Eingang $\mathbf{u}_{R,i}$ der Regelung zurückgeführt, während die approximierten Bewegungen $\hat{\mathbf{x}}_P$ an den Störstellen in den Störstellenausgang $\hat{\mathbf{y}}_S$ ausgegeben werden:

$$\mathbf{u}_{R,i} = \hat{\mathbf{y}}_M := \begin{bmatrix} \hat{\mathbf{x}}_i \\ \hat{\mathbf{n}}_i \end{bmatrix}, \quad \hat{\mathbf{y}}_S := [\hat{\mathbf{x}}_P]. \quad (7.44)$$

Einsetzen der Koppelbedingungen (7.43) und (7.44) in die Differenzialgleichungssysteme in Zustandsdarstellung (7.18) und (7.33) liefert:

$$\dot{\mathbf{z}}_R = \mathbf{A}_R \mathbf{z}_R + \mathbf{B}_{R,s} \mathbf{u}_{R,s} - \mathbf{B}_{R,i} \hat{\mathbf{C}}_M \hat{\mathbf{z}} \quad (7.45)$$

$$\dot{\hat{\mathbf{z}}} = \hat{\mathbf{A}} \hat{\mathbf{z}} + \hat{\mathbf{B}}_A \mathbf{C}_R \mathbf{z}_R + \hat{\mathbf{B}}_S \mathbf{u}_S \quad (7.46)$$

Durch Sortieren der Terme nach Zustands- und Eingangsgrößen und Zusammenfassen der Zustände des regelungs- und elektrotechnischen sowie des reduzierten mechanischen Systems \mathbf{z}_R bzw. $\hat{\mathbf{z}}$ erhält man den Zustandsvektor des Gesamtsystems

$$\mathbf{z}_G := \begin{bmatrix} \mathbf{z}_R \\ \hat{\mathbf{z}} \end{bmatrix} \quad (7.47)$$

und die Systemmatrix \mathbf{A}_G über:

$$\mathbf{A}_G := \begin{bmatrix} \mathbf{A}_R & -\mathbf{B}_{R,i} \hat{\mathbf{C}}_M \\ \hat{\mathbf{B}}_A \mathbf{C}_R & \hat{\mathbf{A}} \end{bmatrix}. \quad (7.48)$$

Die Eingangsgrößen werden in der gleichen Weise im Eingangsvektor

$$\mathbf{u}_G := \begin{bmatrix} \mathbf{u}_{R,s} \\ \hat{\mathbf{u}}_S \end{bmatrix} \quad (7.49)$$

zusammengefasst und die Eingangsmatrix

$$\mathbf{B}_G := \begin{bmatrix} \mathbf{B}_{R,s} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{B}}_S \end{bmatrix} \quad (7.50)$$

gebildet. Damit kann das Differenzialgleichungssystem des mechatronischen Gesamtsystems in der Zustandsdarstellung angegeben werden:

$$\begin{aligned} \dot{\mathbf{z}}_G &= \mathbf{A}_G \mathbf{z}_G + \mathbf{B}_G \mathbf{u}_G \\ \mathbf{y}_G &= \mathbf{C}_G \mathbf{z}_G, \end{aligned} \quad (7.51)$$

mit

$$\mathbf{y}_G := \begin{bmatrix} \hat{\mathbf{y}}_M \\ \hat{\mathbf{y}}_S \end{bmatrix} \quad \text{und} \quad \mathbf{C}_G := \begin{bmatrix} \mathbf{0} & \hat{\mathbf{C}}_M \\ \mathbf{0} & \hat{\mathbf{C}}_S \end{bmatrix}. \quad (7.52)$$

Damit ergibt sich für die Systemgleichungen des geregelten mechatronischen Gesamtsystems das in Abbildung 7.18 dargestellte Blockschaltbild. Die Ein- und Ausgangsvektoren \mathbf{u}_G und \mathbf{y}_G enthalten neben den Führungsgrößen und den Regelgrößen auch Ein- und Ausgänge zur Darstellung

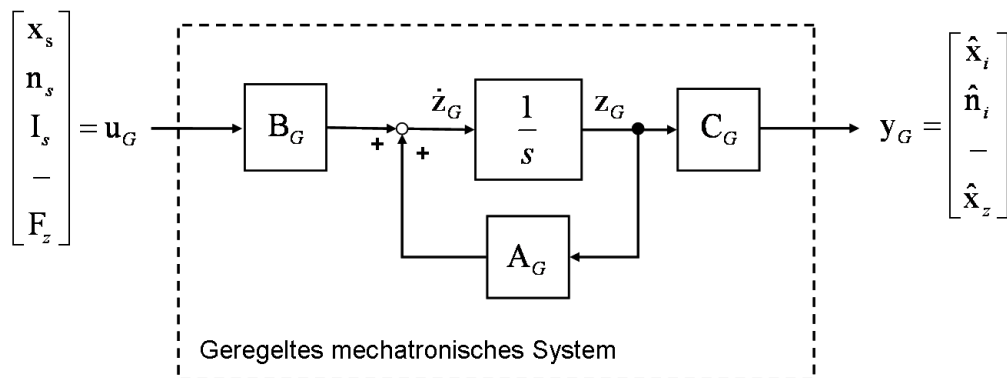


Abbildung 7.18: Zustandsraumdarstellung des geregelten mechatronischen Gesamtsystems nach [138]

des Übertragungsverhaltens in Bezug auf äußere Störeinflüsse. Die Störgrößenschnittstellen \mathbf{F}_P bzw. $\hat{\mathbf{x}}_P$ dienen der Ein- und Ausleitung von Störkräften bzw. Systemantworten an Orten außerhalb der Regelkreise. Über diese kann das Verhalten des mechatronischen Systems in Bezug auf äußere Störkräfte berechnet werden. Dabei müssen für jeden zu berücksichtigenden Störeffekt entsprechende Ein- und Ausgänge an den jeweiligen Orten des mechanischen Systems vorgesehen werden.

Teil III

Modellreduktion in der Simulation von Werkzeugmaschinen

Kapitel 8

Übersicht bekannter Reduktionsmethoden

Die Approximation von Modellen hoher Dimension durch Modelle kleinerer Dimension ist in der Strukturmechanik unverzichtbar. Auch in der Regelungstechnik und der Kontrolltheorie rückte, durch das Anwachsen der behandelten Modelle, die Modellreduktion ins Blickfeld.

In der Vergangenheit hat sich die modale Reduktion und deren Modifikationen als Methode zur Approximation großer Modelle etabliert. In der Strukturmechanik und der Mehrkörperdynamik werden die auf modaler Reduktion basierenden Methoden am häufigsten verwendet. Diese sind meist auch in kommerziellen FEM-Programmen implementiert, z.B. MSC.Patran/Nastran[®],¹ oder Adams[®],². Diese Methoden sind sowohl auf Systeme in Zustandsraumdarstellung als auch auf Systeme zweiter Ordnung der Form (4.1) anwendbar. In Abschnitt 8.3 erfolgt die Darstellung der Methode der modalen Reduktion anhand eines Systems zweiter Ordnung.

Die in Abschnitt 8.1 vorgestellte Methode der Proper Orthogonal Decomposition (POD) ist unabhängig vom Typ des zu reduzierenden Systems. Dies macht diese Methode auch für nichtlineare Systeme anwendbar, z.B. bei der Modellreduktion von Strömungsmodellen oder nichtlinearer Mehrgrößensysteme. Eine Übersicht zur Anwendung der POD auf mechanische Systeme ist u. a. in [106] zu finden.

Aus der Regelungstechnik wurden in jüngerer Zeit verschiedene Ansätze zur Modellreduktion bekannt, welche auf Approximationen von Funktionen mit Hilfe der Singulärwertzerlegung oder dem Momentenabgleich der Übertragungsfunktion basieren. Die ersten Ansätze zur Approximation von Funktionen sind bereits mehr als 200 Jahre alt. Eine für die Modellreduktion wichtige Methode, die Padé-Approximation, geht auf das Jahr 1892 zurück [142]. Zwei weitere für die Modellreduktion durch Momentenabgleich wichtige Algorithmen, der Lanczos- [114] und der Arnoldi-Algorithmus [11] wurden in den 1950-er Jahren veröffentlicht.

Die neuen Verfahren haben den Vorteil, dass sowohl die statischen als auch die dynamischen Eigenschaften des reduzierten Systems gut approximiert werden können. Da die Bewertung auf

¹MSC.Software, Nastran, Kurzform für Nasa Structural Analysis System, von der US-Raumfahrtbehörde NASA entwickeltes Finite-Elemente-Programm.

²Adams/Flex. Mechanical Dynamics Inc.

einer mathematischen Grundlage geschieht, bieten sich zudem Ansätze, die eine Automatisierbarkeit des Reduktionsprozesses ermöglichen.

In Abschnitt 8.2 wird die Methode des balancierten Abschneidens, die als erstes in [133] beschrieben wurde, dargestellt. Die auf dieser Methode basierenden Verfahren generieren reduzierte Systeme mit guten Approximationseigenschaften. Ein großer Vorteil gegenüber anderen Verfahren ist die leichte Bestimmung des Approximationsfehlers des reduzierten Modells. In Hinsicht auf einer automatisierten Modellreduktion stellen die Verfahren basierend auf dieser Methode ideale Reduktionsverfahren dar. Diese Methode wurde ursprünglich für die Reduktion von Modellen in der Zustandsraumdarstellung (3.12) entwickelt [23]. In der letzten Zeit konnte die Anwendung dieser Methode auf Systeme erster Ordnung der Form (3.4) und Systeme zweiter Ordnung der Form (3.1) erweitert werden [170, 171]. Wegen einem hohem Speicherbedarf und aufwendigen Berechnungen ist das Grundverfahren allerdings auf sehr große Modelle nur bedingt anwendbar.

In Abschnitt 8.4 erfolgt eine Darstellung der Methode des Momentenabgleichs. Bei Verfahren mit Momentenabgleich ist es das Ziel, die Eigenschaften der Übertragungsfunktion des Originalsystems durch die Eigenschaften der Übertragungsfunktion des reduzierten Systems zu approximieren. Die wichtigsten Methoden dieser Gruppe sind die Krylov-Unterraum-Methoden, die auch zur Lösung großer Eigenwertprobleme [169] oder zur iterativen Lösung linearer Gleichungssysteme eingesetzt werden, siehe [77, 152]. Diese liefern einen numerisch stabilen Ansatz zur Realisierung des Momentenabgleichs der Übertragungsfunktion eines Systems. Verfahren, die auf diesen Methoden basieren, sind zur Reduktion sehr großer und dünnbesetzter Systeme geeignet und wurden anfänglich für Systeme im Anwendungsbereich der Simulation von elektrischen Schaltkreisen oder mikroelektronischen Systemen (MEMS)³ entwickelt [21, 60, 61, 66, 68]. Die Anwendung dieser Methode auf mechanische Systeme zweiter Ordnung wurden beispielsweise in [155] und [123] beschrieben.

Aufbauend auf den Methoden des vorhergehenden Abschnittes werden in Abschnitt 8.5 Methoden beschrieben, die Systeme in der \mathcal{H}_2 -Norm optimal reduzieren. Dazu wird die Methode der iterativen Krylov-Unterraum-Verfahren und die Approximation durch tangentielle Interpolation beschrieben.

8.1 Proper Orthogonal Decomposition

Eine Methode, die in vielen Anwendungsbereichen zur Anwendung kommt ist die Methode der Proper Orthogonal Decomposition (POD). Diese Methode steht nicht im Mittelpunkt der vorliegenden Arbeit, deshalb sollen an dieser Stelle nur die Grundzüge dieser Methode erläutert werden. Eine detailliertere Übersicht verschiedener Verfahren, die auf dieser Methode basieren, ist beispielsweise in [106] zu finden.

Bei der Proper Orthogonal Decomposition (POD) wird ein Satz von Vektoren $\mathbf{x}_i \in \mathbb{R}^n$ mit $i = 1, \dots, s$ betrachtet, die spaltenweise in einer Matrix

$$\mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_s] \quad (8.1)$$

³Englisch: Micro-Electro-Mechanical Systems

vom Rang $r \leq \min(n, s)$ zusammengefasst sind.

Wird die POD zur Analyse einer Größe $x(t, \mathbf{w})$ mit $t \in \mathbb{R}$ und $\mathbf{w} \in \mathbb{R}^n$ verwendet, so können die Vektoren \mathbf{x}_i als

$$\mathbf{x}_i = \begin{bmatrix} x(t_i, [\mathbf{w}]_1) \\ \vdots \\ x(t_i, [\mathbf{w}]_n) \end{bmatrix} \quad (8.2)$$

definiert werden, wobei die Größen $x(t_i, [\mathbf{w}]_j)$, mit $i = 1, \dots, s$ und $j = 1, \dots, n$ einer endlichen Anzahl von Stichproben s des Feldes $x(t, \mathbf{w})$ entsprechen, siehe [10]. Die Vektoren \mathbf{x}_i werden Snapshots genannt. Die Freiheit bei der Wahl der Snapshots macht die POD zu einem Verfahren mit einem breiten Anwendungsspektrum. Für Systeme in der Zustandsraumdarstellung (3.4) bzw. (3.12) lassen sich Snapshots sowohl im Zeitbereich mit $\mathbf{x}_i = \mathbf{z}(t_i)$ als auch im Frequenzbereich mit $\mathbf{x}_i = \mathbf{Z}(\omega_i)$ definieren.

Das Ziel der POD besteht nun darin, die wesentliche Information, die in der Matrix \mathbf{X} enthalten ist, mit Hilfe eines reduzierten Satzes von Basisvektoren darzustellen. In mathematischer Form wird dieses Ziel durch die Lösung eines Optimierungsproblems erreicht. Dieses Optimierungsproblem besteht darin, einen Satz von Vektoren $\mathbf{u}_i \in \mathbb{R}^n$, mit $i = 1, \dots, \ell$ und $\ell \in [1, s]$ zu finden, der eine Lösung von

$$\max_{\mathbf{u}_1, \dots, \mathbf{u}_\ell} \sum_{j=1}^{\ell} \sum_{i=1}^s (\mathbf{x}_i \cdot \mathbf{u}_j)^2 \quad \text{mit} \quad \mathbf{u}_i^T \cdot \mathbf{u}_j = \delta_{ij}, \quad \text{für} \quad 1 \leq i, j \leq \ell \quad (8.3)$$

darstellt. Theoretisch erhält man die gesuchten Vektoren mit Hilfe des Eigenwertproblems

$$\mathbf{X} \mathbf{X}^T \mathbf{u}_i = \lambda_i \mathbf{u}_i, \quad (8.4)$$

siehe [179]. Da die Matrix $(\mathbf{X} \mathbf{X}^T) \in \mathbb{R}^{n \times n}$, die auch als POD-Kernel (\mathbf{K}_{POD}) bezeichnet wird, aufgrund ihrer Konstruktion positiv semidefinit und symmetrisch ist, besitzt sie nur nicht negative Eigenwerte $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n \geq 0$ sowie n reellwertige Eigenvektoren \mathbf{u}_i , welche sich so normieren lassen, dass sie der Orthogonalitätsbedingung aus (8.3) genügen. Die Eigenvektoren werden auch als Proper Orthogonal Modes (POM) und die normierten Eigenwerte

$$\tilde{\lambda}_i = \frac{\lambda_i}{s} \quad (8.5)$$

als Proper Orthogonal Values (POV) bezeichnet, siehe [106].

Es besteht ein direkter Zusammenhang zwischen dem Eigenwertproblem (8.4) und der in Abschnitt 2.1 beschriebenen Singulärwertzerlegung. Die POM \mathbf{u}_i entsprechen demnach den Linkssingulärvektoren und die POV λ_i den Quadraten der Singulärwerte σ_i der Matrix \mathbf{X} dividiert durch die Mächtigkeit s des betrachteten Datensatzes.

Bei der praktischen Anwendung der POD wird deshalb eine Singulärwertzerlegung der Matrix \mathbf{X} der Form (2.16) angewendet, mit der man die Matrizen $\mathbf{U} \in \mathbb{R}^{n \times n}$, $\mathbf{V} \in \mathbb{R}^{s \times s}$ und $\mathbf{\Sigma} \in \mathbb{R}^{n \times s}$ erhält.

Werden die ersten ℓ POM in einer Matrix

$$\mathbf{U}_\ell = [\mathbf{u}_1 \quad \dots \quad \mathbf{u}_\ell] \quad (8.6)$$

zusammengefasst, so kann mit Hilfe von (2.24) eine Approximation \mathbf{X}_ℓ von \mathbf{X} angegeben werden:

$$\mathbf{X}_\ell = \sum_{i=1}^{\ell} \sigma_i \mathbf{u}_i \mathbf{v}_i = \mathbf{U}_\ell \cdot \underbrace{\boldsymbol{\Sigma}_\ell \cdot (\mathbf{V}_\ell)^T}_{\mathbf{B}_\ell}, \quad (8.7)$$

wobei

$$\mathbf{V}_\ell = [\mathbf{v}_1 \ \cdots \ \mathbf{v}_\ell] \quad (8.8)$$

und

$$\boldsymbol{\Sigma}_\ell = \text{diag}(\sigma_1, \dots, \sigma_\ell). \quad (8.9)$$

Dabei ergibt sich mit (2.35) der in der Frobenius-Norm gemessene Fehler zu

$$\|\mathbf{X} - \mathbf{U}_\ell \cdot \mathbf{B}_\ell\|_F^2 = \sum_{i=\ell+1}^r \sigma_i^2 = \sum_{i=\ell+1}^r \lambda_i. \quad (8.10)$$

Werden für Systeme in der Zustandsraumdarstellung (3.4) bzw. (3.12) Snapshots im Zeitbereich $\mathbf{x}_i = \mathbf{z}(t_i)$ oder im Frequenzbereich $\mathbf{x}_i = \mathbf{Z}(j\omega_i)$ verwendet, so lässt sich mit den Matrizen $\mathbf{V} = \mathbf{U}_\ell \in \mathbb{R}^n$ und $\mathbf{W} = \mathbf{V}$ eine Projektion $\boldsymbol{\Pi} = \mathbf{V}\mathbf{W}^*$ definieren, um ein System mit (5.10) zu reduzieren.

8.2 Balanciertes Abschneiden

Die Approximationsidee bei der Methode des balancierten Abschneidens ist es, diejenigen Zustände zu entfernen, die einerseits viel Steuerenergie benötigen, um angeregt zu werden, und andererseits nur wenig Ausgangsenergie erzeugen.

Deshalb besteht der Ansatz dieser Methode darin, mit Hilfe einer linearen Zustandstransformation eine spezielle Realisierung des Systems zu bestimmen, in der schlecht steuerbare Zustände zugleich schlecht beobachtbare Zustände sind. Diese Realisierung wird balancierte Realisierung genannt. Anschließend können diejenigen Zustände entfernt werden, die einerseits viel Steuerenergie benötigen, um angeregt zu werden und andererseits wenig Ausgangsenergie erzeugen. Die hier wiedergegebene Darstellung der Methode des balancierten Abschneidens orientiert sich an den Darstellungen der Methode in [10, 117].

Definition 8.1 *Die Zustandsraumdarstellung eines steuerbaren, beobachtbaren und stabilen Systems wird balancierte Realisierung genannt, wenn die Gramsche Steuerbarkeitsmatrix (3.117) und die Gramsche Beobachtbarkeitsmatrix (3.118) beide gleich und beide diagonal sind, d.h. wenn gilt*

$$\mathbf{P} = \mathbf{Q} = \boldsymbol{\Sigma} = \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_{2n} \end{bmatrix} \quad (8.11)$$

wobei σ_k reelle und positive Werte mit

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{2n} > 0 \quad (8.12)$$

die Hankel-Singulärwerte (3.131) des Systems sind.

Um die balancierte Realisierung eines Systems der Form (3.12) zu erhalten, wird eine lineare Zustandsraumtransformation $\mathbf{z} = \mathbf{T} \cdot \tilde{\mathbf{z}}$ durchgeführt, wobei $\mathbf{T} \in \mathbb{K}^{2n \times 2n}$ eine reguläre zeitinvariante Matrix ist. Das transformierte System ergibt sich dann zu

$$\begin{aligned}\dot{\tilde{\mathbf{z}}}(t) &= \tilde{\mathbf{A}}\tilde{\mathbf{z}}(t) + \tilde{\mathbf{B}}\mathbf{u}(t), \\ \mathbf{y}(t) &= \tilde{\mathbf{C}}\tilde{\mathbf{z}}(t),\end{aligned}\tag{8.13}$$

wobei

$$\tilde{\mathbf{A}} = \mathbf{T}^{-1}\mathbf{A}\mathbf{T}, \quad \tilde{\mathbf{B}} = \mathbf{T}^{-1}\mathbf{B} \quad \text{und} \quad \tilde{\mathbf{C}} = \mathbf{C}\mathbf{T}.\tag{8.14}$$

Für die Gramschen Matrizen in den neuen Koordinaten ergibt sich

$$\tilde{\mathbf{P}} = \mathbf{T}^{-1}\mathbf{P}\mathbf{T}^{-*} \quad \text{und} \quad \tilde{\mathbf{Q}} = \mathbf{T}^*\mathbf{Q}\mathbf{T}.\tag{8.15}$$

Unter der linearen Koordinatentransformation (8.15) sind die Eigenwerte der Matrizen \mathbf{P} und \mathbf{Q} also nicht invariant. Allerdings folgt für die Eigenwerte λ_i des Produktes der beiden Matrizen

$$\lambda_i(\mathbf{P}\mathbf{Q}) = \lambda_i(\mathbf{T}^{-1}\mathbf{P}\mathbf{Q}\mathbf{T}) = \lambda_i(\tilde{\mathbf{P}}\tilde{\mathbf{Q}}),\tag{8.16}$$

so dass diese unabhängig von der Wahl des Zustandes sind und als Invarianten des Ein- und Ausgangsverhalten angesehen werden können [10, 188].

Sind die symmetrischen und positiv definiten Matrizen \mathbf{P} und \mathbf{Q} die eindeutigen Lösungen der Gleichungen (3.119) bzw. (3.120), dann sind die transformierten, symmetrischen und positiv definiten Gramschen Matrizen $\tilde{\mathbf{P}} = \mathbf{T}^{-1}\mathbf{P}\mathbf{T}^{-*}$ und $\tilde{\mathbf{Q}} = \mathbf{T}^*\mathbf{Q}\mathbf{T}$ die eindeutigen Lösungen der Lyapunov-Gleichungen

$$\tilde{\mathbf{A}}\tilde{\mathbf{P}} + \tilde{\mathbf{P}}\tilde{\mathbf{A}}^* + \tilde{\mathbf{B}}\tilde{\mathbf{B}}^* = \mathbf{0},\tag{8.17}$$

$$\tilde{\mathbf{A}}^*\tilde{\mathbf{Q}} + \tilde{\mathbf{Q}}\tilde{\mathbf{A}} + \tilde{\mathbf{C}}^*\tilde{\mathbf{C}} = \mathbf{0},\tag{8.18}$$

siehe [10].

In [10, 188] wird gezeigt, dass es die Transformationseigenschaften (8.15) erlauben, für ein asymptotisch stabiles, steuerbares und beobachtbares System der Form (3.12) eine Zustandstransformation anzugeben, so dass in den neuen Koordinaten gilt

$$\tilde{\mathbf{P}} = \tilde{\mathbf{Q}} = \mathbf{\Sigma}\tag{8.19}$$

wobei $\mathbf{\Sigma} := \text{diag}(\sigma_i)$.

Lemma 8.1 (Zustandsraumtransformation \mathbf{T}) *Gegeben sei das steuerbare, beobachtbare und stabile System der Form (3.12) sowie die entsprechenden Gramschen Matrizen \mathbf{P} und \mathbf{Q} . Dann ist durch*

$$\mathbf{T} = \mathbf{\Sigma}^{1/2}\mathbf{K}^*\mathbf{U}^{-1} \quad \text{und} \quad \mathbf{T}^{-1} = \mathbf{U}\mathbf{K}\mathbf{\Sigma}^{-1/2},\tag{8.20}$$

wobei \mathbf{U} die Faktor-Matrix der Cholesky-Zerlegung

$$\mathbf{P} = \mathbf{U}\mathbf{U}^*\tag{8.21}$$

ist und die Matrizen \mathbf{K} und $\mathbf{\Sigma} = \text{diag}(\sigma_1, \dots, \sigma_{2n})$ aus der Eigenwertwertzerlegung der positiv definiten Matrix $\mathbf{U}^* \mathbf{Q} \mathbf{U}$

$$\mathbf{U}^* \mathbf{Q} \mathbf{U} = \mathbf{K} \mathbf{\Sigma}^2 \mathbf{K}^* \quad (8.22)$$

resultieren, eine balancierende Zustandstransformation gegeben.

BEWEIS: Siehe [10]. □

Diese Darstellung wird balancierte Realisierung bezeichnet. Die entsprechende Zustandstransformation \mathbf{T} kann mit Algorithmus 1 berechnet werden [10].

Algorithmus 1 Algorithmus zur Balancierung von Systemen [10]

Eingabe: Matrizen \mathbf{P} und \mathbf{Q}

Ausgabe: Zustandstransformation \mathbf{T} und \mathbf{T}^{-1}

- 1: function $[\mathbf{T}, \mathbf{T}^{-1}] = \text{Balancierung}(\mathbf{P}, \mathbf{Q})$
 - 2: Cholesky-Zerlegung $\mathbf{P} = \mathbf{U} \mathbf{U}^*$
 - 3: Eigenwertzerlegung: $\mathbf{U}^* \mathbf{Q} \mathbf{U} = \mathbf{K} \mathbf{\Sigma}^2 \mathbf{K}^*$
 - 4: Transformationsmatrizen: $\mathbf{T} = \mathbf{\Sigma}^{1/2} \mathbf{K}^* \mathbf{U}^{-1}$ und $\mathbf{T}^{-1} = \mathbf{U} \mathbf{K} \mathbf{\Sigma}^{-1/2}$
-

In der balancierten Realisierung des Systems sind die schlecht steuerbaren Zustände nun zugleich auch schlecht beobachtbare Zustände.

Ist \mathbf{T} als eine, das System (3.12) balancierende, Transformation mit

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > \sigma_{r+1} \geq \dots \geq \sigma_{2n} > 0 \quad (8.23)$$

gegeben, dann kann die Gramsche Matrix $\mathbf{\Sigma}$ in (8.20) entsprechend partitioniert werden:

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_2 \end{bmatrix}, \quad (8.24)$$

wobei $\mathbf{\Sigma}_1 \in \mathbb{K}^{r \times r}$.

Das bedeutet, dass das System in zwei Subsysteme partitioniert werden kann, wobei ein Subsystem die größten und das andere die kleinen Hankel-Singulärwerte enthält.

Partitioniert man die Systemmatrizen entsprechend

$$\tilde{\mathbf{A}} = \begin{bmatrix} \tilde{\mathbf{A}}_{11} & \tilde{\mathbf{A}}_{12} \\ \tilde{\mathbf{A}}_{21} & \tilde{\mathbf{A}}_{22} \end{bmatrix}, \quad \tilde{\mathbf{B}} = \begin{bmatrix} \tilde{\mathbf{B}}_1 \\ \tilde{\mathbf{B}}_2 \end{bmatrix}, \quad \text{und} \quad \mathbf{C} = \begin{bmatrix} \tilde{\mathbf{C}}_1 & \tilde{\mathbf{C}}_2 \end{bmatrix}, \quad (8.25)$$

so kann das System in die Subsysteme $(\tilde{\mathbf{A}}_{ii}, \tilde{\mathbf{B}}_i, \tilde{\mathbf{C}}_i)$, $i = 1, 2$ aufgeteilt werden, wobei $\tilde{\mathbf{A}}_{11} \in \mathbb{K}^{r \times r}$, $\tilde{\mathbf{B}}_1 \in \mathbb{K}^{r \times m}$ und $\tilde{\mathbf{C}}_1 \in \mathbb{K}^{q \times r}$.

Man erhält ein reduziertes System, indem man die Matrizen des zweiten Subsystems vernachlässigt. Mit dem Originalsystem sind auch die beiden Subsysteme asymptotisch stabil, sowie steuer- und beobachtbar. Für das System $(\tilde{\mathbf{A}}_{11}, \tilde{\mathbf{B}}_1, \tilde{\mathbf{C}}_1)$ gilt zudem die Fehlerabschätzung

$$\|\tilde{\mathbf{H}} - \tilde{\mathbf{H}}_1\|_{\mathcal{H}_\infty} \leq 2(\sigma_{r+1} + \dots + \sigma_{2n}), \quad (8.26)$$

wobei $\tilde{\mathbf{H}}$ die Übertragungsfunktion des balancierten Originalsystems und $\tilde{\mathbf{H}}_1$ die Übertragungsfunktion des reduzierten Systems darstellen [10].

Definition 8.2 (Balanciertes Abschneiden) Ist (8.25) ein stabiles, steuerbares und beobachtbares System in balancierter Realisierung, dann wird das Subsystem $(\tilde{\mathbf{A}}_{11}, \tilde{\mathbf{B}}_1, \tilde{\mathbf{C}}_1)$ mit $\tilde{\mathbf{A}}_{11} \in \mathbb{K}^{r \times r}$, $\tilde{\mathbf{B}}_1 \in \mathbb{K}^{r \times m}$ und $\tilde{\mathbf{C}}_1 \in \mathbb{K}^{q \times r}$ durch balanciertes Abschneiden reduziertes System der Ordnung r genannt.

Um das Subsystem $(\tilde{\mathbf{A}}_{11}, \tilde{\mathbf{B}}_1, \tilde{\mathbf{C}}_1)$ zu bestimmen, ist es nicht notwendig, das gesamte System zu balancieren. In [86] wird auf das in Abschnitt 5.2 beschriebene Projektionsverfahren zurückzugreifen, um ein Verfahren mit weniger Rechenaufwand zu erhalten. Dabei wird das reduzierte System durch Anwendung einer schiefen Projektion mit Hilfe zweier Matrizen \mathbf{V} und \mathbf{W} generiert. Dieses Verfahren wird durch den Algorithmus 2 angegeben.

Algorithmus 2 Square Root Algorithmus mit Abschneiden [10, 86]

Eingabe: Matrizen \mathbf{P} und \mathbf{Q}

Ausgabe: Projektionsmatrizen \mathbf{V} und \mathbf{W}

- 1: function $[\mathbf{V}, \mathbf{W}] = \text{Balanciertes_Abschneiden}(\mathbf{P}, \mathbf{Q})$
 - 2: Cholesky-Zerlegung $\mathbf{P} = \mathbf{U} \mathbf{U}^*$
 - 3: Eigenwertzerlegung: $\mathbf{U}^* \mathbf{Q} \mathbf{U} = \mathbf{K} \mathbf{\Sigma}^2 \mathbf{K}^*$
 - 4: Partitionierung: $\mathbf{K} = [\mathbf{K}_1 \ \mathbf{K}_2]$, $\mathbf{\Sigma} = \text{diag}(\mathbf{\Sigma}_1 \ \mathbf{\Sigma}_2)$,
 - 5: Transformationsmatrizen: $\mathbf{T} = \mathbf{\Sigma}_1^{-1/2} \mathbf{K}_1^* \mathbf{U}^{-1}$ und $\mathbf{T}^{-1} = \mathbf{U} \mathbf{K}_1 \mathbf{\Sigma}_1^{-1/2}$
-

Mit dem Originalsystem sind auch die beiden Subsysteme asymptotisch stabil, sowie steuer- und beobachtbar. Für das System $(\tilde{\mathbf{A}}_{11}, \tilde{\mathbf{B}}_1, \tilde{\mathbf{C}}_1)$ kann zudem eine Fehlerabschätzung angegeben werden [10].

Satz 8.1 Sei ein asymptotisch stabiles, steuerbares und beobachtbares LTI System $\check{\mathbf{\Sigma}}_{1o}$ der Form (3.12) gegeben. Das durch balanciertes Abschneiden bestimmte reduzierte System $\hat{\mathbf{\Sigma}}_{1o}$ hat die folgenden Eigenschaften:

1. Das reduzierte System $\hat{\mathbf{\Sigma}}_{1o}$ ist balanciert und asymptotisch stabil.
2. Seien σ_i , $i = 1, \dots, 2n$ die Singulärwerte von $\check{\mathbf{\Sigma}}_{1o}$ mit Vielfachheiten m_i und habe $\hat{\mathbf{\Sigma}}_{1o}$ die Singulärwerte σ_i , $i = 1, \dots, r$ mit $r < 2n$ und Vielfachheiten m_i . Der Approximationsfehler in der \mathcal{H}_∞ -Norm des reduzierten Systems ist dann gegeben durch:

$$\|\check{\mathbf{\Sigma}}_{1o} - \hat{\mathbf{\Sigma}}_{1o}\|_{\mathcal{H}_\infty} \leq 2(\sigma_{r+1} + \dots + \sigma_{2n}). \quad (8.27)$$

BEWEIS: Siehe [10]. □

Das hier dargestellte Verfahren für Systeme erster Ordnung der Form (3.12) kann für die Reduktion von Systemen in anderen Darstellungsformen angepasst werden. Für die Reduktion von Systemen erster Ordnung in der allgemeinen Zustandsdarstellung der Form (3.4) sei auf [170, 171] verwiesen. In [146] wird eine Methode vorgestellt, die Systeme zweiter Ordnung der Form (3.1) mit der Methode des balancierten Abschneidens reduziert.

Die Methode des balancierten Abschneidens ist in Hinsicht auf die Stabilitäts- und die globale Fehlerbestimmung des reduzierten Systems ein ideales Reduktionsverfahren. Allerdings

sind die bisher entwickelten Verfahren des balancierten Abschneidens in der Praxis noch nicht auf sehr große Modelle anwendbar, da die Lösung von zwei Lyapunov-Gleichungen und zusätzlich eine Singulärwertzerlegung zur Bestimmung des reduzierten Systems benötigt werden, wodurch ein hoher Speicher- und Berechnungsaufwand notwendig ist. Um diese Methode auch auf große Modelle anwenden zu können, wird deshalb auf Approximationsverfahren zurückgegriffen, siehe dazu [10, 19, 23, 24, 100, 143, 144].

8.3 Modale Reduktion

Der Ausgangspunkt bei der modalen Reduktion ist die Reihendarstellung der Übertragungsfunktion (4.31) des modal transformierten Systems (4.15). Die Approximationsidee besteht darin, die nach der Eigenfrequenz geordnete ($\omega_1 \leq \omega_2 \leq \dots \leq \omega_n$) Reihe (4.31) nach dem r -ten Glied abzurechnen:

$$\hat{H}(j\omega) = \sum_{i=1}^r \frac{[\bar{C}_p + j\omega \bar{C}_v]_{*i} \cdot [\bar{F}]_{i*}}{\omega_i^2 - \omega^2 + 2j\omega \omega_i D_{L,i}}. \quad (8.28)$$

Die Anzahl r der Glieder wird dabei so groß gewählt, dass die Eigenfrequenz ω_{r+1} genügend weit von der Erregerfrequenz ω entfernt liegt. In der Literatur findet man dazu verschiedene Empfehlungen, bis zu welchem Wert der Erregerfrequenz die Eigenmoden berücksichtigt werden sollten. In [73] findet man beispielsweise die Empfehlung, alle Eigenmoden bis zum doppelten Wert der maximal auftretenden Erregerfrequenz zu berücksichtigen.

Aus der Reihendarstellung (4.31) ist zu ersehen, dass die Größe der einzelnen Reihenglieder nicht nur von den Eigenwerten des Systems, sondern auch von \bar{C}_p , \bar{C}_v und \bar{F} abhängen. Somit kann eine Eigenmode einen großen Einfluss auf das Übertragungsverhalten haben, obwohl deren Eigenfrequenz relativ weit von der Erregerfrequenz entfernt liegt.

Wird durch die Vernachlässigung des i -ten Gliedes der Reihe (4.31) das Übertragungsverhalten des Systems stark beeinflusst, bezeichnet man die Eigenfrequenz und die zugehörige Eigenmode als dominant. Aus diesem Grund besteht das Hauptproblem der modalen Reduktion darin, die dominanten Eigenmoden zu bestimmen.

In [74] wird gezeigt, dass proportional gedämpfte Systeme mit kleiner Dämpfung näherungsweise balanciert sind und es somit möglich ist, zur i -ten Eigenmode mit

$$\tilde{\sigma}_i = \frac{\|[\bar{C}_p + j\omega \bar{C}_v]_{*i}\|_2 \cdot \|[\bar{F}]_{i*}\|_2}{4\omega_i D_{L,i}} \quad (8.29)$$

einen entsprechenden approximierten Hankel-Singulärwert $\tilde{\sigma}_i$ anzugeben. Auf diese Weise ist es möglich, den einzelnen Eigenmoden Beiträge zur \mathcal{H}_∞ -Norm (8.27) zuzuordnen, wodurch sich dann eine Approximation des Abschneidefehlers für das reduzierte System bezüglich dieser Normen angeben lässt. Mit dieser Methode lassen sich Eigenmoden auswählen, die einen wichtigen Einfluss auf das Übertragungsverhalten des Systems haben, doch werden häufig viele Momente benötigt, um eine zufriedenstellende Approximation zu erhalten. Die Ursache dafür liegt darin, dass in den Eigenmoden keine Information bezüglich der räumlichen Verteilung der wirkenden Kräfte enthalten ist, siehe dazu [73, 131, 185]. Die Approximationsgüte kann erhöht werden, indem in der Reihe (4.31) nur Anteile von Eigenmoden verwendet werden, von denen ein wichtiger

Beitrag zur Übertragungsfunktion erwartet wird. Zusätzlich werden die nicht berücksichtigten Moden mit Hilfe eines Durchgriffsterms \mathbf{R} näherungsweise erfasst. Damit erhält man die approximierte Übertragungsfunktion:

$$\hat{\mathbf{H}}(\omega) = \sum_{i=1}^r \frac{[\bar{\mathbf{C}}_p + \omega \bar{\mathbf{C}}_v]_{*i} \cdot [\bar{\mathbf{F}}]_{i*}}{\omega_i^2 - \omega^2 + 2\omega \omega_i D_{L,i}} + \mathbf{R}. \quad (8.30)$$

Bei der sogenannten *Mode Acceleration* Methode [45, 46, 3] wird der Term \mathbf{R} so gewählt, dass die stationäre Lösung korrekt abgebildet wird, d.h. $\hat{\mathbf{H}}(0) = \mathbf{H}(0)$.

Durch eine Ergänzung der modalen Basis um Korrekturmoden, die es erlauben, bestimmte Eigenschaften des Originalmodells im reduzierten Modell zu erhalten, kann eine Verbesserung der Approximation erreicht werden. In der Strukturmechanik werden solche Ansätze bei der Kopplung von Substrukturen verwendet. Mit der Verwendung von *Attachment Modes* oder *Constraint Modes* wird z. B. der statische Einfluss von Lasten berücksichtigt [46, 93].

Sogenannte *Inertia Relief Attachment Modes* [46, 93] werden verwendet, wenn eine elastische Struktur auch Starrkörpermoden beinhaltet.

Die Verwendung von *Frequency Response Modes* stellt eine Verallgemeinerung dieser Verfahren dar [50]. Hierbei wird die Amplitude der stationären Lösung eines harmonisch erregten mechanischen Systems als Korrekturmoden verwendet, die sich für die Erregerfrequenzen ω_{FRM} bei vernachlässigter Dämpfung zu

$$\Phi_{FRM} := (-\omega_{FRM}^2 \mathbf{M} + \mathbf{K})^{-1} \cdot \mathbf{F}, \quad (8.31)$$

ergibt, siehe z. B. [3]. Die *Frequency Response Modes* gehen für eine Anregungsfrequenz $\omega_{FRM} = 0$ im Falle einer statisch gelagerten Struktur in die *Attachment Modes* über. Liegen ungelagerte Strukturen vor, so sind nur Werte $\omega_{FRM} \neq 0$ zulässig. Für kleine Werte von ω_{FRM} besteht jedoch ein enger Zusammenhang mit den *Inertia Relief Attachment Modes* [50].

Bei der Verwendung von Eigen- und Korrekturmoden besteht ein kritischer Punkt darin, dass die durch ggf. linear abhängige Spaltenvektoren erweiterte Matrix (numerisch) nicht vollen Spaltenrang besitzt. Eine Methode dies zu verhindern, besteht in der Transformation der Korrekturmoden [46, 50, 185], so dass die resultierenden Moden nur noch Anteile aus den vernachlässigten Eigenmoden enthalten und massenorthogonal bezüglich der berücksichtigten Eigenmoden sind. Diese Moden werden als *Residual Attachment Modes* bezeichnet.

Der Aspekt linear abhängiger Spaltenvektoren deutet auf einen Defizit dieser Verfahren hin. Ein nicht voller Spaltenrang der Projektionsmatrix bedeutet, dass nicht alle Korrekturmoden zur Verbesserung der Approximation beitragen. Bei Verwendung der *Frequency Response Modes* stellt sich zudem die Frage, wie viele und welche Anregungskreisfrequenzen ω_{FRM} für ein bestimmtes Anregungsspektrum sinnvoll sind.

8.4 Krylov-Unterraum-Verfahren

Bei der Modellreduktion mit Krylov-Unterräumen handelt es sich um Projektionsverfahren, die auf einer Interpolation der Übertragungsfunktion mit Hilfe eines Momentenabgleiches basieren.

Die Methoden des Momentenabgleiches werden auch Moment-Matching⁴-Methoden genannt. Bei diesen Methoden wird gefordert, dass eine bestimmte Anzahl an Momenten des Originalsystems mit den Momenten des reduzierten Systems übereinstimmen.

Im ersten Abschnitt dieses Kapitels wird das grundlegende Prinzip des Momentenabgleich näher dargestellt. Im darauf folgenden Abschnitt 8.4.2 wird zunächst auf die Methode des expliziten Momentenabgleiches eingegangen. Die Methoden zum expliziten Momentenabgleich sind nur für kleine reduzierte Systeme numerisch stabil zu berechnen, was deren Verwendung in vielen Anwendungsgebieten einschränkt.

Ein numerisch stabiler Ansatz reduzierte Modelle, die den Anforderungen des Momentenabgleiches genügen, zu erzeugen, ist die Verwendung von Projektionsmatrizen, deren Spaltenvektoren Basen von geeigneten Krylov-Unterräumen darstellen. Mit der Verwendung dieser Projektionsmatrizen ist es möglich, die geforderte Eigenschaft des Momentenabgleiches implizit zu erhalten. In Abschnitt 8.4.3 wird deshalb auf den Zusammenhang der Eigenschaft des Momentenabgleiches der Übertragungsfunktion eines reduzierten Systems erster Ordnung und Krylov-Unterräumen eingegangen. Anschließend werden Möglichkeiten vorgestellt, wie Basen für geeignete Krylov-Unterräume numerisch stabil generiert werden können.

Im darauf folgenden Abschnitt 8.4.4 werden Verfahren zur Modellreduktion von Systemen zweiter Ordnung auf Basis von Krylov-Unterräumen vorgestellt und die Sonderfälle von ungedämpften und gedämpften Systemen diskutiert.

Ein System in Zustandsdarstellung, das durch Transformation eines Systems zweiter Ordnung entstanden ist, besitzt nach Abschnitt 3.1 eine charakteristische Blockstruktur. Wird eine geeignete Krylov-Unterraummethode auf das Modell in Zustandsdarstellung angewendet, so überträgt sich diese Struktur in gewisser Weise auf die Blöcke der Krylov-Sequenz. Das Ausnutzen dieser Eigenschaft erlaubt es, eine direkte Methode für Systeme zweiter Ordnung zu entwickeln. Dazu werden in [14, 24, 68, 122, 123, 155] geeignete Verfahren vorgestellt und diskutiert. Für das Anwendungsgebiet der Strukturmechanik sind die Verfahren, die in [14, 122, 155] vorgeschlagen werden, am besten geeignet, da mit diesen die Struktur des Systems als System zweiter Ordnung erhalten bleibt. Bei diesen Verfahren werden die Projektionsräume durch sogenannte Krylov-Unterräume zweiter Ordnung beschrieben. Geeignete Methoden zur stabilen Berechnung der Basen dieser Räume findet man in [15, 123, 155].

Mechanische Systeme liegen häufig als ungedämpfte oder als proportional gedämpfte Systeme vor. Es ist daher sinnvoll, die daraus resultierenden Auswirkungen auf die Krylov-Unterraum-Verfahren zu untersuchen. Zu diesem Zweck können die Arbeiten [54, 20] herangezogen werden, die die Auswirkungen einer Rayleigh-Dämpfung auf die Krylov-Unterraum-Verfahren als Sonderfall der proportionalen Dämpfung untersuchen.

Die Darstellungen in diesem Abschnitt orientieren sich an [10, 111, 117, 155].

⁴Dieser Begriff ist in der englischen Fachliteratur für den Begriff *Momentenabgleich* gebräuchlich.

8.4.1 Momentenabgleich

Da das Prinzip des Momentenabgleiches auf Systeme anderer Darstellungsformen und auf Systeme zweiter Ordnung übertragen werden kann, wird das grundlegende Konzept des Momentenabgleiches hier ausschließlich anhand eines Systems erster Ordnung der Form (3.4) dargestellt. Die Potenzreihenentwicklung um die Entwicklungsstelle $s_0 \in \mathbb{C}$ der Übertragungsfunktion eines Systems der Form (3.4) ist durch

$$\mathbf{H}(s) = \sum_{j=0}^{\infty} -\mathbf{C}((\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{E})^j(\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{B}(s - s_0)^j = \sum_{j=0}^{\infty} \mathbf{H}_j(s_0)(s - s_0)^j \quad (8.32)$$

gegeben. Die Koeffizienten $\mathbf{H}_j(s_0)$ bezeichnen dabei die Momente der Übertragungsfunktion, siehe dazu auch die Definition 3.10 in Abschnitt 3.2.3. Bei den Methoden basierend auf Momentenabgleich wird nun gefordert, dass eine bestimmte Anzahl der ersten Momente der Übertragungsfunktion $\hat{\mathbf{H}}(s)$ des reduzierten Systems mit den ersten Momenten der Übertragungsfunktion des Originalsystems $\mathbf{H}(s)$ übereinstimmen. Bei den mit diesen Methoden reduzierten Modellen handelt es sich um sogenannte Padé-Modelle. Man sagt auch, dass die Übertragungsfunktion des reduzierten Systems $\hat{\mathbf{H}}(s)$ eine Padé-Approximation der Übertragungsfunktion $\mathbf{H}(s)$ an der Entwicklungsstelle s_0 ist.

Definition 8.3 (Padé-Modell) *Ein reduziertes Modell der Dimension r wird r -tes Padé Modell an der Entwicklungsstelle s_0 genannt, wenn die Potenzreihenentwicklung an der Stelle s_0 der Übertragungsfunktion $\mathbf{H}(s)$ mit der Potenzreihenentwicklung der Übertragungsfunktion des reduzierten Systems $\hat{\mathbf{H}}(s)$ an möglichst vielen führenden Termen übereinstimmt, d.h. wenn gilt*

$$\mathbf{H}(s) = \hat{\mathbf{H}}(s) + \mathcal{O}((s - s_0)^{p(r)}), \quad (8.33)$$

wobei $p(r)$ so groß wie möglich ist.

8.4.2 Expliziter Momentenabgleich

Die folgende Herleitung der Methode des expliziten Momentenabgleiches, basiert auf der Darstellung in [38], wobei zur einfacheren Darstellung ein SISO-System betrachtet wird. In diesem Fall handelt es sich bei der Übertragungsfunktion $H(s)$ um eine skalarwertige Funktion. Das grundlegende Konzept des expliziten Momentenabgleiches kann auf allgemeine MIMO-Systeme übertragen werden. Bei der r -ten Padé-Approximation $\hat{H}(s)$ um die Entwicklungsstelle s_0 , für die die Gleichung (8.33) mit $p = 2r$ gilt, handelt es sich um eine skalarwertige rationale Funktion mit Polynomen Φ und Ψ im Zähler bzw. Nenner, die einen maximalen Grad $r - 1$ bzw. r aufweisen:

$$\hat{H}(s) = \frac{\Phi_{r-1}(s - s_0)}{\Psi_r(s - s_0)} = \frac{\phi_{r-1}(s - s_0)^{r-1} + \dots + \phi_1(s - s_0) + \phi_0}{\psi_r(s - s_0)^r + \psi_{r-1}(s - s_0)^{r-1} + \dots + \psi_1(s - s_0) + 1}, \quad (8.34)$$

wobei $\psi_0 = 1$ gewählt wurde.

Bei der Betrachtung eines SISO-Systems erhält man nach Multiplikation beider Seiten der Gleichung (8.33) von rechts mit $\Psi_r(s - s_0)$ die skalarwertige Funktion

$$H(s)\Psi_r(s - s_0) = \Phi_{n-1}(s - s_0) + \mathcal{O}((s - s_0)^{2r})\Psi_r(s - s_0). \quad (8.35)$$

Nach der Definition 3.10 kann die Übertragungsfunktion $H(s)$ an der Stelle s als Potenzreihe um die Entwicklungsstelle s_0 geschrieben werden:

$$H(s) = \sum_{j=0}^{\infty} H_j \cdot (s - s_0)^j. \quad (8.36)$$

Mit den Gleichungen (8.35) und (8.36) erhält man die Gleichung

$$\begin{aligned} & \left(\sum_{j=0}^{2r-1} H_j \cdot (s - s_0)^j + \sum_{j=2r}^{\infty} H_j \cdot (s - s_0)^j \right) \left(\sum_{j=1}^r \psi_j (s - s_0)^j + 1 \right) = \\ & = \sum_{j=0}^{r-1} \phi_j (s - s_0)^j + \mathcal{O}((s - s_0)^{2r}) \left(\sum_{j=1}^r \psi_j (s - s_0)^j + 1 \right), \end{aligned} \quad (8.37)$$

die auch als

$$\begin{aligned} & H_0 + (H_0\psi_1 + H_1)(s - s_0) + (H_0\psi_2 + H_1\psi_1 + H_2)(s - s_0)^2 + \dots \\ & + (H_0\psi_{r-1} + H_1\psi_{r-2} + \dots + H_{r-2}\psi_1 + H_{r-1})(s - s_0)^{r-1} \\ & + (H_0\psi_r + H_1\psi_{r-1} + \dots + H_{r-1}\psi_1 + H_r)(s - s_0)^r \\ & + (H_1\psi_r + H_2\psi_{r-1} + \dots + H_r\psi_1 + H_{r+1})(s - s_0)^{r+1} + \dots \\ & + (H_{r-1}\psi_r + H_r\psi_{r-1} + \dots + H_{2r-2}\psi_1 + H_{2r-1})(s - s_0)^{2r-1} = \\ & \phi_0 + \phi_1(s - s_0) + \phi_2(s - s_0)^2 + \dots + \phi_{r-1}(s - s_0)^{r-1} \\ & + \left(\mathcal{O}((s - s_0)^{2r}) - \sum_{j=2r}^{\infty} H_j (s - s_0)^j \right) \left(\sum_{j=1}^r \psi_j (s - s_0)^j + 1 \right) \end{aligned} \quad (8.38)$$

geschrieben werden kann.

Durch Vergleich der Terme $(s - s_0)^j$, $j = r, \dots, 2r - 1$ der Gleichung (8.38) können die Koeffizienten $\{\psi_j\}$ des Polynoms $\Psi_r(s - s_0)$ als Gleichungssystem geschrieben werden:

$$\begin{bmatrix} H_0 & H_1 & \dots & H_{r-2} & H_{r-1} \\ H_1 & H_2 & \dots & H_{r-1} & H_r \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ H_{r-2} & H_{r-1} & \dots & H_{2r-4} & H_{2r-3} \\ H_{r-1} & H_r & \dots & H_{2r-3} & H_{2r-2} \end{bmatrix} \begin{bmatrix} \psi_r \\ \psi_{r-1} \\ \vdots \\ \psi_2 \\ \psi_1 \end{bmatrix} = - \begin{bmatrix} H_r \\ H_{r+1} \\ \vdots \\ H_{2r-2} \\ H_{2r-1} \end{bmatrix}. \quad (8.39)$$

Die Koeffizienten $\{\phi_j\}$ können durch den Vergleich der $(s - s_0)^j$, $j = 1, \dots, r - 1$ Terme der Gleichung (8.38) bestimmt werden:

$$\begin{aligned} \phi_0 &= H_0 \\ \phi_1 &= H_0\psi_1 + H_1 \\ &\vdots \\ \phi_{r-1} &= H_0\psi_{r-1} + H_1\psi_{r-2} + \dots + H_{r-2}\psi_1 + H_{r-1}. \end{aligned} \quad (8.40)$$

Mit (8.39) und (8.40) ist es also möglich, die $2r$ freien Parameter ψ_1, \dots, ψ_r und $\phi_0, \dots, \phi_{r-1}$ von $\hat{H}(s)$ so zu bestimmen, dass die ersten $2r$ Momente um die Entwicklungsstelle s_0 übereinstimmen. Dieser Ansatz wird *asymptotic waveform evaluation (AWE)* genannt [38]. Allerdings ist dieses Verfahren zur Bestimmung von Padé-Approximationen wegen der expliziten Berechnung der Momente nur für sehr kleine ($r \leq 10$) reduzierte Modelle numerisch stabil verwendbar, siehe [67, 81].

Um eine explizite Berechnung der Momente zu umgehen, werden deshalb implizite Verfahren zur Berechnung von Padé-Approximationen verwendet. Dabei sind die Krylov-Unterraum-Verfahren die wichtigsten Vertreter dieser Verfahren.

8.4.3 Krylov-Unterraum-Verfahren erster Ordnung

Die Potenzreihenentwicklung (8.32) der Übertragungsfunktion $\mathbf{H}(s)$ um die Entwicklungsstelle $s_0 = 0$ eines Systems der Form (3.4) kann mit $\tilde{\mathbf{A}} = \mathbf{A}^{-1}\mathbf{E}$ und $\tilde{\mathbf{B}} = -\mathbf{A}^{-1}\mathbf{B}$ auch geschrieben werden als

$$\mathbf{H}(s) = \mathbf{C}(\mathbf{I} + s\tilde{\mathbf{A}} + s^2\tilde{\mathbf{A}}^2 + \dots)\tilde{\mathbf{B}} = \sum_{j=0}^{\infty} \mathbf{H}_j(0) s^j, \quad (8.41)$$

wobei $\mathbf{H}_j(0) = \mathbf{C}\tilde{\mathbf{A}}^j\tilde{\mathbf{B}} = \mathbf{C}(\mathbf{A}^{-1}\mathbf{E})^j(-\mathbf{A}^{-1})\mathbf{B}$, $j = 0, 1, \dots$ den Momenten (3.46) der Übertragungsfunktion $\mathbf{H}(s)$ des Systems entsprechen.

Alternativ kann die Potenzreihenentwicklung der Übertragungsfunktion auch mit $\tilde{\mathbf{A}} = \mathbf{E}\mathbf{A}^{-1}$ und $\tilde{\mathbf{C}} = -\mathbf{C}\mathbf{A}^{-1}$ geschrieben werden als

$$\mathbf{H}(s) = \tilde{\mathbf{C}}(\mathbf{I} + s\tilde{\mathbf{A}} + s^2\tilde{\mathbf{A}}^2 + \dots)\mathbf{B} = \sum_{j=0}^{\infty} \mathbf{H}_j(0) s^j, \quad (8.42)$$

wobei $\mathbf{H}_j(0) = \tilde{\mathbf{C}}\tilde{\mathbf{A}}^j\mathbf{B} = -\mathbf{C}\mathbf{A}^{-1}(\mathbf{E}\mathbf{A}^{-1})^j\mathbf{B}$, $j = 0, 1, \dots$ den Momenten (3.46) der Übertragungsfunktion $\mathbf{H}(s)$ des Systems entsprechen.

Betrachtet man die Potenzreihenentwicklungen der Form (8.41) oder (8.42), so wird die Verbindung zwischen Krylov-Unterräumen und den Momenten der Übertragungsfunktion eines Systems deutlich. Die Elemente der Krylov-Sequenz sind mit den Eingangs- oder Ausgangsmomenten der Übertragungsfunktion identisch, wobei die Eingangsmomente als $\mathbf{H}_j^E(0) = \tilde{\mathbf{A}}^j\tilde{\mathbf{B}}$ und die Ausgangsmomente als $\mathbf{H}_j^A(0) = (\tilde{\mathbf{A}}^*)^j\tilde{\mathbf{C}}^*$ mit $\tilde{\mathbf{C}} = -\mathbf{A}^{-*}\mathbf{C}^*$ definiert sind. Dadurch können durch die Verwendung von Krylov-Unterraum-Methoden die Momente der Übertragungsfunktion eines reduzierten Systems mit den Momenten der Übertragungsfunktion des Originalsystems implizit zur Übereinstimmung gebracht werden.

Damit das reduzierte System bestimmte gewünschte Eigenschaften des Momentenabgleiches erfüllt, ist eine geeignete Wahl der Matrizen zur Erzeugung einer Basis des Krylov-Unterraumes entscheidend. Die gewünschten Eigenschaften des Momentenabgleiches werden bei einem SISO-System erster Ordnung durch die Wahl der Matrizen \mathbf{P} und \mathbf{q} in (2.48) bzw. bei einem MIMO-System durch \mathbf{P} und \mathbf{Q} in (2.50) festgelegt. Werden die Matrizen dabei so gewählt, dass die Eingangsmomente $\mathbf{H}_j^E(0)$ zur Erzeugung einer Basis des Krylov-Unterraumes verwendet werden, so spricht man von einem Eingangs-Krylov-Unterraum. Werden hingegen die Ausgangsmomente

$\mathbf{H}_j^A(0)$ zur Erzeugung einer Basis des Krylov-Unterraumes verwendet werden, so spricht man von einem Ausgangs-Krylov-Unterraum.

Die Krylov-Unterraum-Verfahren lassen sich somit in die zwei Gruppen der einseitigen und zweiseitigen Verfahren unterteilen.

Definition 8.4 (Ein- oder zweiseitige Verfahren) *Krylov-Unterraum-Verfahren, bei denen entweder der Eingangs- oder der Ausgangs-Krylov-Unterraum verwendet wird, werden einseitige Verfahren genannt.*

Krylov-Unterraum-Verfahren, bei denen sowohl der Eingangs- als auch der Ausgangs-Krylov-Unterraum verwendet wird, werden zweiseitige Verfahren genannt.

Die Wahl geeigneter Matrizen \mathbf{P} und \mathbf{q} bzw. \mathbf{Q} wird im folgenden für unterschiedlich gewählte Entwicklungsstellen dargestellt. Die Darstellung der folgenden Sätze stützt sich dabei auf die entsprechenden Sätze in [71, 81, 155, 176].

8.4.3.1 Verfahren mit $s_0 = 0$

Für Verfahren, die eine Entwicklungsstelle $s_0 = 0$ verwenden, werden die Projektionsmatrizen \mathbf{V} und \mathbf{W} so bestimmt, dass die Momente um die Entwicklungsstelle $s_0 = 0$ des reduzierten Systems mit den ersten Momenten der Übertragungsfunktion des originalen Systems übereinstimmen. Liegt das System in der Form (3.12) vor, erhält man die zu den folgenden Darstellungen äquivalenten Aussagen durch Austausch der Matrix \mathbf{E} durch die Einheitsmatrix \mathbf{I} .

Wählt man die Projektionsmatrizen $\mathbf{V}, \mathbf{W} \in \mathbb{K}^{2n \times r}$ für das Projektionsverfahren (5.10) aus Abschnitt 5.2 mit

$$\begin{aligned} \mathbf{P} &= \mathbf{A}^{-1} \mathbf{E} \quad \text{und} \\ \mathbf{q} &= -\mathbf{A}^{-1} \mathbf{b} \quad \text{bzw.} \quad \mathbf{Q} = -\mathbf{A}^{-1} \mathbf{B} \end{aligned} \tag{8.43}$$

in (2.48) bzw. (2.50) so, dass gilt

$$\begin{aligned} \mathcal{K}_k(\mathbf{A}^{-1} \mathbf{E}, -\mathbf{A}^{-1} \mathbf{b}) &\subseteq \mathcal{V} = \text{colspan}(\mathbf{V}) \quad \text{bzw.} \\ \mathcal{K}_k(\mathbf{A}^{-1} \mathbf{E}, -\mathbf{A}^{-1} \mathbf{B}) &\subseteq \mathcal{V} = \text{colspan}(\mathbf{V}), \\ \mathbf{W} &= \mathbf{V} \end{aligned} \tag{8.44}$$

und die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist, dann erhält man ein einseitiges Verfahren, das auf der Berechnung einer Basis des sogenannten (SISO-) Eingangs-Krylov-Unterraumes bzw. des sogenannten (MIMO-) Eingangs-Krylov-Unterraum basiert. Alternativ kann auch der Ausgangs-Krylov-Unterraum zum Bestimmen geeigneter Spaltenvektoren der Projektionsmatrizen \mathbf{V} und \mathbf{W} verwendet werden. Dazu wählt man die Projektionsmatrizen $\mathbf{V}, \mathbf{W} \in \mathbb{K}^{2n \times r}$ für das Projektionsverfahren (5.10) aus Abschnitt 5.2 mit

$$\begin{aligned} \mathbf{P} &= \mathbf{A}^{-*} \mathbf{E}^* \quad \text{und} \\ \mathbf{q} &= -\mathbf{A}^{-*} \mathbf{c}^* \quad \text{bzw.} \quad \mathbf{Q} = -\mathbf{A}^{-*} \mathbf{C}^* \end{aligned} \tag{8.45}$$

in (2.48) bzw. (2.50) so, dass gilt

$$\begin{aligned}\mathcal{K}_k(\mathbf{A}^{-*} \mathbf{E}^*, -\mathbf{A}^{-*} \mathbf{c}^*) &\subseteq \mathcal{W} = \text{colspan}(\mathbf{W}) \quad \text{bzw.} \\ \mathcal{K}_k(\mathbf{A}^{-*} \mathbf{E}^*, -\mathbf{A}^{-*} \mathbf{C}^*) &\subseteq \mathcal{W} = \text{colspan}(\mathbf{W}), \\ \mathbf{V} &= \mathbf{W}\end{aligned}\tag{8.46}$$

und die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist, dann erhält man ein einseitiges Verfahren, das auf der Berechnung einer Basis des sogenannten (SISO-) Ausgangs-Krylov-Unterraumes bzw. des sogenannten (MIMO-) Ausgangs-Krylov-Unterraum basiert.

Anmerkung 8.1 In (8.44) bzw. (8.46) wird $\mathbf{V} = \mathbf{W}$ verwendet, wobei $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist. Um die Eigenschaften des Momentenabgleiches zu erfüllen können jedoch beliebige Matrizen \mathbf{V} bzw. \mathbf{W} verwendet werden, für die gilt, dass $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist.

Für den Momentenabgleich reduzierter Systeme, die mit diesen einseitigen Verfahren erzeugt werden, gilt die Aussage des folgenden Satzes,

Satz 8.2 (Einseitiger Momentenabgleich an der Stelle $s_0 = 0$) Sei ein System erster Ordnung der Form (3.4) gegeben. Bilden die Spaltenvektoren einer Projektionsmatrix $\mathbf{V} \in \mathbb{K}^{2n \times r}$ bzw. einer Projektionsmatrix $\mathbf{W} \in \mathbb{K}^{2n \times r}$ eine Basis des Krylov-Unterraumes $\mathcal{K}_k(\mathbf{P}, \mathbf{q})$ mit \mathbf{P} und \mathbf{q} bzw. \mathbf{Q} wie in (8.43) bzw. wie in (8.45), und wird die Projektionsmatrix \mathbf{W} bzw. \mathbf{V} so gewählt, dass die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist, dann stimmen mindestens die ersten $\lfloor r/m \rfloor$ bzw. die ersten $\lfloor r/q \rfloor$ Momente der Übertragungsfunktion des mit der Projektion $\mathbf{\Pi} = \mathbf{V} \mathbf{W}^*$ erzeugten reduzierten Systems und des originalen Systems an der Entwicklungsstelle $s_0 = 0$ überein.

BEWEIS: Siehe z. B. [81, 155, 176]. □

Wählt man für das Projektionsverfahren die Projektionsmatrix \mathbf{V} wie in (8.44) und die Projektionsmatrix \mathbf{W} wie in (8.46), so erhält man ein zweiseitiges Verfahren. Der folgende Satz macht eine Aussage bezüglich des Momentenabgleiches eines mit Hilfe dieses Verfahrens reduzierten Systems.

Satz 8.3 (Zweiseitiger Momentenabgleich an der Stelle $s_0 = 0$) Sei ein System erster Ordnung der Form (3.4) gegeben. Bilden die Spaltenvektoren einer Projektionsmatrix $\mathbf{V} \in \mathbb{K}^{2n \times r}$ eine Basis des Krylov-Unterraumes $\mathcal{K}_k(\mathbf{P}, \mathbf{q})$ mit \mathbf{P} und \mathbf{q} bzw. \mathbf{Q} wie in (8.43) bzw. des Block-Krylov-Unterraumes $\mathcal{K}_k(\mathbf{P}, \mathbf{Q})$ und bilden die Spaltenvektoren einer Projektionsmatrix $\mathbf{W} \in \mathbb{K}^{2n \times r}$ eine Basis des Krylov-Unterraumes $\mathcal{K}_k(\mathbf{P}, \mathbf{q})$ mit \mathbf{P} und \mathbf{q} bzw. \mathbf{Q} wie in (8.45) und ist die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär, dann stimmen mindestens die ersten $\lfloor r/m \rfloor + \lfloor r/q \rfloor$ Momente der Übertragungsfunktion des mit der Projektion $\mathbf{\Pi} = \mathbf{V} \mathbf{W}^*$ erzeugten reduzierten Systems und des originalen Systems an der Entwicklungsstelle $s_0 = 0$ überein.

BEWEIS: Siehe z. B. [71, 81, 155, 176]. □

8.4.3.2 Verfahren mit $s_0 \in \mathbb{C}$

Die Methode des Momentenabgleiches kann auf eine beliebige Entwicklungsstelle $s_0 \in \mathbb{C}$ erweitert werden. Dazu werden die Projektionsmatrizen \mathbf{V} und \mathbf{W} so bestimmt, dass die Momente

um eine beliebige Entwicklungsstelle $s_0 \in \mathbb{C}$ nach Definition 3.10 des reduzierten Systems mit den ersten Momenten der Übertragungsfunktion des originalen Systems übereinstimmen. Liegt das System in der Form (3.12) vor, erhält man die zu den folgenden Darstellungen äquivalenten Aussagen durch Austausch der Matrix \mathbf{E} durch die Einheitsmatrix \mathbf{I} .

Bei Verfahren, die den Eingangs-Krylov-Unterraum verwenden, werden die Projektionsmatrizen \mathbf{V} und \mathbf{W} für das Projektionsverfahren (5.10) aus Abschnitt 5.2 mit

$$\begin{aligned} \mathbf{P} &= (\mathbf{A} - s_0 \mathbf{E})^{-1} \mathbf{E} \quad \text{und} \\ \mathbf{q} &= -(\mathbf{A} - s_0 \mathbf{E})^{-1} \mathbf{b} \quad \text{bzw.} \quad \mathbf{Q} = -(\mathbf{A} - s_0 \mathbf{E})^{-1} \mathbf{B} \end{aligned} \quad (8.47)$$

in (2.48) bzw. (2.50) so gewählt, dass gilt

$$\begin{aligned} \mathcal{K}_k((\mathbf{A} - s_0 \mathbf{E})^{-1} \mathbf{E}, -(\mathbf{A} - s_0 \mathbf{E})^{-1} \mathbf{b}) &\subseteq \mathcal{V} = \text{colspan}(\mathbf{V}) \quad \text{bzw.} \\ \mathcal{K}_k((\mathbf{A} - s_0 \mathbf{E})^{-1} \mathbf{E}, -(\mathbf{A} - s_0 \mathbf{E})^{-1} \mathbf{B}) &\subseteq \mathcal{V} = \text{colspan}(\mathbf{V}), \\ \mathbf{W} &= \mathbf{V} \end{aligned} \quad (8.48)$$

und die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist.

Bei Verfahren, die den Ausgangs-Krylov-Unterraum verwenden, werden die Projektionsmatrizen \mathbf{V} und \mathbf{W} für das Projektionsverfahren (5.10) aus Abschnitt 5.2 mit

$$\begin{aligned} \mathbf{P} &= (\mathbf{A} - s_0 \mathbf{E})^{-*} \mathbf{E}^* \quad \text{und} \\ \mathbf{q} &= -(\mathbf{A} - s_0 \mathbf{E})^{-*} \mathbf{c}^* \quad \text{bzw.} \quad \mathbf{Q} = -(\mathbf{A} - s_0 \mathbf{E})^{-*} \mathbf{C}^* \end{aligned} \quad (8.49)$$

in (2.48) bzw. (2.50) so gewählt, dass gilt

$$\begin{aligned} \mathcal{K}_k((\mathbf{A} - s_0 \mathbf{E})^{-*} \mathbf{E}^*, -(\mathbf{A} - s_0 \mathbf{E})^{-*} \mathbf{c}^*) &\subseteq \mathcal{W} = \text{colspan}(\mathbf{W}) \quad \text{bzw.} \\ \mathcal{K}_k((\mathbf{A} - s_0 \mathbf{E})^{-*} \mathbf{E}^*, -(\mathbf{A} - s_0 \mathbf{E})^{-*} \mathbf{C}^*) &\subseteq \mathcal{W} = \text{colspan}(\mathbf{W}), \\ \mathbf{V} &= \mathbf{W} \end{aligned} \quad (8.50)$$

und die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist.

Für den Momentenabgleich eines mit einem einseitigen Verfahren reduzierten Systems mit Entwicklungsstelle $s_0 \in \mathbb{C}$ gilt die Aussage des folgenden Satzes.

Satz 8.4 (Einseitiger Momentenabgleich an der Stelle $s_0 \in \mathbb{C}$) *Sei ein System erster Ordnung der Form (3.4) gegeben. Bilden die Spaltenvektoren einer Projektionsmatrix $\mathbf{V} \in \mathbb{K}^{2n \times r}$ oder $\mathbf{W} \in \mathbb{K}^{2n \times r}$ eine Basis des Krylov-Unterraumes $\mathcal{K}_k(\mathbf{P}, \mathbf{q})$ mit \mathbf{P} und \mathbf{q} bzw. \mathbf{Q} wie in (8.47) bzw. wie in (8.49) und wird die Projektionsmatrix \mathbf{W} bzw. \mathbf{V} so gewählt, dass die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist, dann stimmen mindestens die ersten $\lfloor r/m \rfloor$ bzw. die ersten $\lfloor r/q \rfloor$ Momente der Übertragungsfunktion des mit der Projektion $\mathbf{\Pi} = \mathbf{V} \mathbf{W}^*$ erzeugten reduzierten Systems und des originalen Systems an der Entwicklungsstelle $s_0 \in \mathbb{C}$ überein.*

BEWEIS: Siehe z. B. [81, 155, 176]. □

Wählt man für das Projektionsverfahren die Projektionsmatrix \mathbf{V} wie in (8.48) und die Projektionsmatrix \mathbf{W} wie in (8.50), so erhält man ein zweiseitiges Verfahren. Der folgende Satz macht eine Aussage bezüglich des Momentenabgleiches eines mit Hilfe dieses Verfahrens reduzierten Systems.

Satz 8.5 (Zweiseitiger Momentenabgleich an der Stelle $s_0 \in \mathbb{C}$) Sei ein System erster Ordnung der Form (3.4) gegeben. Bilden die Spaltenvektoren einer Projektionsmatrix $\mathbf{V} \in \mathbb{K}^{2n \times r}$ eine Basis des Krylov-Unterraumes $\mathcal{K}_k(\mathbf{P}, \mathbf{q})$ mit \mathbf{P} und \mathbf{q} bzw. \mathbf{Q} wie in (8.47) und bilden die Spaltenvektoren einer Projektionsmatrix $\mathbf{W} \in \mathbb{K}^{2n \times r}$ eine Basis des Krylov-Unterraumes $\mathcal{K}_k(\mathbf{P}, \mathbf{q})$ mit \mathbf{P} und \mathbf{q} bzw. \mathbf{Q} wie in (8.49) und ist die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär, dann stimmen mindestens die ersten $\lfloor r/m \rfloor + \lfloor r/q \rfloor$ Momente der Übertragungsfunktion des mit der Projektion $\mathbf{\Pi} = \mathbf{V} \mathbf{W}^*$ erzeugten reduzierten Systems und des originalen Systems an der Entwicklungsstelle $s_0 \in \mathbb{C}$ überein.

BEWEIS: Siehe [71, 81, 155, 176]. □

8.4.3.3 Verfahren mit mehr als einer Entwicklungsstelle

Die Anwendung von Verfahren, die eine Entwicklungsstelle verwenden, führt zu reduzierten Systemen, deren Übertragungsfunktion die Übertragungsfunktion des originalen Systems in der Umgebung der Entwicklungsstelle gut approximieren. Soll ein breiter Frequenzbereich gut approximiert werden, kann die beschriebene Methode des Momentenabgleiches auf mehrere Entwicklungsstellen $s_i \in \mathbb{C}$, $i = 1, 2, \dots, \hat{i}$ erweitert werden.

Bei Verfahren, die den Eingangs-Krylov-Unterraum verwenden, werden die Projektionsmatrizen \mathbf{V} und \mathbf{W} für das Projektionsverfahren (5.10) aus Abschnitt 5.2 für $i = 1, \dots, \hat{i}$ mit

$$\begin{aligned} \mathbf{P} = \mathbf{P}_i &= (\mathbf{A} - s_i \mathbf{E})^{-1} \mathbf{E} \quad \text{und} \\ \mathbf{q} = \mathbf{q}_i &= -(\mathbf{A} - s_i \mathbf{E})^{-1} \mathbf{b} \quad \text{bzw.} \quad \mathbf{Q} = \mathbf{Q}_i = -(\mathbf{A} - s_i \mathbf{E})^{-1} \mathbf{B} \end{aligned} \quad (8.51)$$

in (2.48) bzw. (2.50) so gewählt, dass gilt

$$\begin{aligned} \bigcup_{i=1}^{\hat{i}} \mathcal{K}_k((\mathbf{A} - s_i \mathbf{E})^{-1} \mathbf{E}, -(\mathbf{A} - s_i \mathbf{E})^{-1} \mathbf{b}) &\subseteq \mathcal{V} = \text{colspan}(\mathbf{V}) \quad \text{bzw.} \\ \bigcup_{i=1}^{\hat{i}} \mathcal{K}_k((\mathbf{A} - s_i \mathbf{E})^{-1} \mathbf{E}, -(\mathbf{A} - s_i \mathbf{E})^{-1} \mathbf{B}) &\subseteq \mathcal{V} = \text{colspan}(\mathbf{V}), \end{aligned} \quad (8.52)$$

$$\mathbf{W} = \mathbf{V}$$

und die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist. Bei Verfahren, die den Ausgangs-Krylov-Unterraum verwenden, werden die Projektionsmatrizen \mathbf{V} und \mathbf{W} für das Projektionsverfahren (5.10) aus Abschnitt 5.2 für $i = 1, \dots, \hat{i}$ mit

$$\begin{aligned} \mathbf{P} = \mathbf{P}_i &= (\mathbf{A} - s_i \mathbf{E})^{-*} \mathbf{E}^* \quad \text{und} \\ \mathbf{q} = \mathbf{q}_i &= -(\mathbf{A} - s_i \mathbf{E})^{-*} \mathbf{c}^* \quad \text{bzw.} \quad \mathbf{Q} = \mathbf{Q}_i = -(\mathbf{A} - s_i \mathbf{E})^{-*} \mathbf{C}^* \end{aligned} \quad (8.53)$$

in (2.48) bzw. (2.50) so gewählt, dass gilt

$$\begin{aligned} \bigcup_{i=1}^{\hat{i}} \mathcal{K}_k((\mathbf{A} - s_i \mathbf{E})^{-*} \mathbf{E}^*, -(\mathbf{A} - s_i \mathbf{E})^{-*} \mathbf{c}^*) &\subseteq \mathcal{W} = \text{colspan}(\mathbf{W}) \quad \text{bzw.} \\ \bigcup_{i=1}^{\hat{i}} \mathcal{K}_k((\mathbf{A} - s_i \mathbf{E})^{-*} \mathbf{E}^*, -(\mathbf{A} - s_i \mathbf{E})^{-*} \mathbf{C}^*) &\subseteq \mathcal{W} = \text{colspan}(\mathbf{W}), \end{aligned} \quad (8.54)$$

$$\mathbf{V} = \mathbf{W}$$

und die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist.

Satz 8.6 (Einseitiger Momentenabgleich an einer Stelle $s_i \in \mathbb{C}$) *Sei ein System erster Ordnung der Form (3.4) gegeben. Bilden die Spaltenvektoren einer Projektionsmatrix $\mathbf{V} \in \mathbb{K}^{2n \times r}$ oder einer Projektionsmatrix $\mathbf{W} \in \mathbb{K}^{2n \times r}$ eine Basis von $\bigcup_{i=1}^{\hat{i}} \mathcal{K}_k(\mathbf{P}_i, \mathbf{q}_i)$ bzw. von $\bigcup_{i=1}^{\hat{i}} \mathcal{K}_k(\mathbf{P}_i, \mathbf{Q}_i)$ mit \mathbf{P}_i und \mathbf{q}_i bzw. \mathbf{Q}_i wie in (8.51) bzw. wie in (8.53) und wird die Projektionsmatrix \mathbf{W} bzw. \mathbf{V} so gewählt, dass die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist, dann stimmen mindestens die ersten $\lfloor r/(m \cdot \hat{i}) \rfloor$ bzw. die ersten $\lfloor r/(q \cdot \hat{i}) \rfloor$ Momente der Übertragungsfunktion des mit der Projektion $\mathbf{\Pi} = \mathbf{V} \mathbf{W}^*$ erzeugten reduzierten Systems und des originalen Systems an der Entwicklungsstelle $s_i \in \mathbb{C}$, $i = 1, 2, \dots, \hat{i}$ überein.*

BEWEIS: Siehe z. B. [81, 155, 176]. □

Wählt man für das Projektionsverfahren die Projektionsmatrix \mathbf{V} wie in (8.52) und die Projektionsmatrix \mathbf{W} wie in (8.54), so erhält man ein zweiseitiges Verfahren mit mehreren Entwicklungsstellen $s_i \in \mathbb{C}$, $i = 1, 2, \dots, \hat{i}$. Der folgende Satz macht eine Aussage bezüglich des Momentenabgleiches eines mit einem zweiseitigen Verfahren reduzierten Systems.

Satz 8.7 (Zweiseitiger Momentenabgleich an einer Stelle $s_0 \in \mathbb{C}$) *Sei ein System erster Ordnung der Form (3.4) gegeben. Bilden die Spaltenvektoren einer Projektionsmatrix $\mathbf{V} \in \mathbb{K}^{2n \times r}$ eine Basis von $\bigcup_{i=1}^{\hat{i}} \mathcal{K}_k(\mathbf{P}_i, \mathbf{q}_i)$ bzw. von $\bigcup_{i=1}^{\hat{i}} \mathcal{K}_k(\mathbf{P}_i, \mathbf{Q}_i)$ mit \mathbf{P}_i und \mathbf{q}_i bzw. \mathbf{Q}_i wie in (8.51) und bilden die Spaltenvektoren einer Projektionsmatrix $\mathbf{W} \in \mathbb{K}^{2n \times r}$ eine Basis von $\bigcup_{i=1}^{\hat{i}} \mathcal{K}_k(\mathbf{P}_i, \mathbf{q}_i)$ bzw. von $\bigcup_{i=1}^{\hat{i}} \mathcal{K}_k(\mathbf{P}_i, \mathbf{Q}_i)$ mit \mathbf{P}_i und \mathbf{q}_i bzw. \mathbf{Q}_i wie in (8.53) und ist die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär, dann stimmen mindestens die ersten $\lfloor r/(m \cdot \hat{i}) \rfloor + \lfloor r/(q \cdot \hat{i}) \rfloor$ Momente der Übertragungsfunktion des mit der Projektion $\mathbf{\Pi} = \mathbf{V} \mathbf{W}^*$ erzeugten reduzierten Systems und des originalen Systems an der Entwicklungsstelle $s_i \in \mathbb{C}$, $i = 1, 2, \dots, \hat{i}$ überein.*

BEWEIS: Siehe z. B. [71, 81, 155, 176]. □

In einigen Anwendungsgebieten ist es notwendig, bestimmte Eigenschaften wie z. B. die Stabilität oder die Passivität des Originalsystems auf das reduzierte System zu übertragen. In [66, 68, 70] werden Krylov-Unterraum-Verfahren vorgestellt, die dies garantieren.

Um die Basis eines Krylov-Unterraumes berechnen zu können, wird ein numerisch stabiler Algorithmus benötigt. Zu diesem Zweck stehen verschiedene Algorithmen zur Auswahl, die unterschiedliche Basen des betrachteten Krylov-Unterraumes konstruieren. Der folgende Satz besagt, dass die Wahl der Basis des verwendeten Krylov-Unterraumes dabei keine Auswirkungen auf die Übertragungsfunktion des reduzierten Systems hat.

Satz 8.8 (Invarianz der Übertragungsfunktion bzgl. der Krylov-Basen) *Die Übertragungsfunktion $\hat{\mathbf{H}}(s)$ eines reduzierten Systems, dass durch ein zweiseitiges Verfahren oder durch ein einseitiges Verfahren mit Projektionsmatrizen \mathbf{V} und \mathbf{W} für die $\mathbf{W} = \mathbf{V}$ gilt, erzeugt wurde, ist unabhängig von der Wahl der Basis des Eingangs- bzw. Ausgangs-Krylov-Unterraumes.*

BEWEIS: Siehe [155]. □

8.4.3.4 Numerische Aspekte

Reduktionsverfahren, die auf Krylov-Unterräumen basieren, sind für Berechnungen mit sehr großen Systemen gut geeignet, da diese zur Berechnung der Spaltenvektoren der Projektionsmatrizen \mathbf{V} und \mathbf{W} nur innere Produkte und Matrix-Vektor-Multiplikationen benötigen.

Bei der Implementierung der Krylov-Unterraum-Verfahren ist zur Berechnung der Spaltenvektoren der Projektionsmatrizen \mathbf{V} und \mathbf{W} die direkte Berechnung der Krylov-Sequenz nach der Definition des Krylov-Unterraumes (2.48) bzw. (2.50) mit geeignetem Matrizen \mathbf{P} und \mathbf{Q} nicht ratsam. In [67] wird gezeigt, dass die Blöcke der Krylov-Sequenz

$$\{ \mathbf{Q}, \mathbf{P}\mathbf{Q}, \mathbf{P}^2\mathbf{Q}, \dots, \mathbf{P}^{r-1}\mathbf{Q} \} \quad (8.55)$$

sehr schnell gegen den Unterraum der dominanten Eigenwerte der Matrix \mathbf{P} konvergieren. Bei endlicher Genauigkeit enthalten die Blöcke dann nur noch Informationen über diese Unterräume. Aus diesem Grund werden Algorithmen benötigt, die die Basen der Krylov-Unterräume numerisch robust erzeugen.

Ein solcher Algorithmus ist der (unsymmetrische) Block-Lanczos-Algorithmus, der simultan zwei Basen \mathbf{V} und \mathbf{W} erzeugt, die der Biorthogonalitätsbedingung $\mathbf{W}^*\mathbf{V} = \mathbf{I}$ genügen. Die Dimension r der mit Hilfe dieses Algorithmus erzeugten Basen \mathbf{V} und \mathbf{W} wird durch die Anzahl \tilde{m} und der Anzahl \tilde{q} der linear unabhängigen Spalten der Eingabematrizen \mathbf{Q} und $\tilde{\mathbf{Q}}$ sowie der gewählten Stufe des Krylov-Unterraumes k mit $r = \max\{\tilde{m}, \tilde{q}\} \cdot k$ festgelegt.

Ein weiterer wichtiger Algorithmus ist der Block-Arnoldi-Algorithmus, der eine Basis \mathbf{V} erzeugt, die der Orthogonalitätsbedingung $\mathbf{V}^*\mathbf{V} = \mathbf{I}$ genügt. Werden zwei Basen \mathbf{V} und \mathbf{W} benötigt, so können diese mit diesem Algorithmus unabhängig voneinander erzeugt werden. Die Dimension r der mit Hilfe dieses Algorithmus erzeugten Basis wird durch die Anzahl \tilde{m} der linear unabhängigen Spalten der Eingangsmatrix \mathbf{Q} und der gewählten Stufe des Krylov-Unterraumes k mit $r = \tilde{m} \cdot k$ festgelegt.

Ein Aspekt, der bei der Erzeugung von Block-Krylov-Unterräumen berücksichtigt werden muss, ist die lineare Abhängigkeit von Spaltenvektoren des k -ten Blockes $\mathbf{P}^{k-1}\mathbf{Q}$ der Krylov-Sequenz (8.55) von Spaltenvektoren in Blöcken, die vorher erzeugt wurden. Ein linear abhängiger Spaltenvektor in einem Block der Krylov-Sequenz führt dazu, dass alle weiteren Vektoren in den Blöcken der Krylov-Sequenz, die mit dem linear abhängigen Spaltenvektor berechnet werden, ebenfalls linear abhängig zu bereits berechneten Basisvektoren sind. Während eine Krylov-Sequenz mit nur einem Startvektor \mathbf{q} an der Stelle abgebrochen werden kann, an der die lineare Abhängigkeit auftritt, ist dies bei der Block-Version nicht möglich. Da nicht zu erwarten ist, dass die lineare Abhängigkeit für alle Spalten des k -ten Blockes gleichzeitig auftritt, würde so auch die Krylov-Sequenz für nicht linear abhängige Spaltenvektoren des Blockes abgebrochen. Wünschenswert ist somit eine Methode, die die Berechnung der Krylov-Sequenz für alle bis auf die linear abhängigen Spaltenvektoren weiter durchführt.

In [6] wird dazu vorgeschlagen, anstelle der Krylov-Sequenz (8.55) die Sequenz

$$\{ \mathbf{Q}_1, \mathbf{P}\mathbf{Q}_2, \mathbf{P}^2\mathbf{Q}_3, \dots, \mathbf{P}^{r-1}\mathbf{Q}_r \} \quad (8.56)$$

zu verwenden, wobei $\mathbf{Q}_1 = \mathbf{Q}$ und \mathbf{Q}_k für jedes $k = 2, \dots, r$ eine Submatrix von \mathbf{Q}_{k-1} ist, die man durch Streichen derjenigen Spalten von \mathbf{Q}_{k-1} erhält, aus denen bei der Auswertung von $\mathbf{P}^{k-2} \mathbf{Q}_{k-1}$ linear abhängige Einträge resultieren. Dieser Vorgang wird exakte Deflation genannt. Bei numerischen Berechnungen ist es zusätzlich notwendig, zu bereits berechneten Basisvektoren annähernd linear abhängige Basisvektoren zu entfernen, was durch eine sogenannte inexakte Deflation erreicht wird [6]. Lanczos-Algorithmen zur Erzeugung von Basen von Block-Krylov-Unterräumen unter Berücksichtigung von Deflation werden z. B. in [6, 155] vorgestellt. Dem entsprechende Arnoldi-Algorithmen können z. B. [67, 155] entnommen werden.

Algorithmus 3 gibt den Ablauf des Arnoldi-Algorithmus und Algorithmus 4 gibt den Ablauf des Lanczos-Algorithmus in Pseudocode wieder. Die Algorithmen basieren auf dem unsymmetrischen Lanczos-Algorithmus, und dem Arnoldi-Algorithmus, wie sie in [10] dargestellt werden. Die Algorithmen werden hier mit Deflation mit Deflationstoleranz τ wie in [155] wiedergegeben.

Algorithmus 3 Arnoldi-Algorithmus nach [10, 155]

Eingabe: Matrizen \mathbf{P}, \mathbf{Q} ; Stufe des Block-Krylov-Unterraumes k ; Deflationstoleranz τ

Ausgabe: Basis \mathbf{V}

```

1: function  $\mathbf{V} = \text{Arnoldi}(\mathbf{P}, \mathbf{Q}, k, \tau)$ 
2: Entfernung aller linear abhängigen Spaltenvektoren der Matrix  $\mathbf{Q}$  ▷ Initialisierungen
3:  $\mathbf{V} = []$ 
4: for  $j = 1 : k$ 
5:   if ( $j == 1$ )
6:      $\mathbf{R}_j = \mathbf{Q}$  ▷ Berechnung des ersten Blockes
7:   else
8:      $\mathbf{R}_j = \mathbf{P} \mathbf{R}_{j-1}$  ▷ Berechnung der folgenden Blöcke
9:   end if
10:  for  $i = 1 : \text{Spaltenanzahl}(\mathbf{R}_j)$  ▷ Orthonormalisierung
11:    for  $t = 1 : \text{Spaltenanzahl}(\mathbf{V})$  ▷ Orthogonalisierung
12:       $h_{j+1,j} = [\mathbf{R}_j]_{:,i}^* [\mathbf{V}]_{:,t}$ 
13:       $[\mathbf{R}_j]_{:,i} = [\mathbf{R}_j]_{:,i} - h_{j+1,j} [\mathbf{V}]_{:,t}$ 
14:    end for
15:     $h_{j,j} = ||[\mathbf{R}_j]_{:,i}||_2$ 
16:    if  $h_{j,j} \geq \tau$ 
17:       $\mathbf{v}_i = \frac{1}{h_{j,j}} [\mathbf{R}_j]_{:,i}$  ▷ Normierung
18:       $\mathbf{V} = [\mathbf{V} \ \mathbf{v}_i]$ 
19:    else
20:      Deflation
21:    end if
22:  end for
23: end for
```

8.4.3.5 Rationale Krylov-Verfahren

Bei der Modellreduktion mit den sogenannten rationalen Krylov-Verfahren werden mehr als eine Entwicklungsstelle verwendet. Aus diesem Grund werden Basen für die Vereinigung mehrerer Eingangs- bzw. Ausgangs-Block-Krylov-Unterräume benötigt. Diese lassen sich durch Erweiterungen der Block-Versionen des Lanczos- oder des Arnoldi-Algorithmus numerisch stabil erzeugen.

Algorithmus 4 (unsymmetrischer) Block-Lanczos-Algorithmus mit Deflation nach [10, 155]

Eingabe: Matrizen $P, \check{P}, Q, \check{Q}$; Stufe des Block-Krylov-Unterraumes k ; Deflationstoleranz τ
Ausgabe: Basen V und W

```

1: function [  $V, W$  ] = Lanczos( $P, \check{P}, Q, \check{Q}, k, \tau$ )
2: Entfernung aller linear abhängigen Spaltenvektoren der Matrizen  $Q$  und  $\check{Q}$  ▷ Initialisierungen
3:  $\tilde{m} = \text{Spaltenanzahl}(Q)$ ;  $\tilde{q} = \text{Spaltenanzahl}(\check{Q})$ 
4:  $\hat{k} = k \cdot \max \{ \tilde{m}, \tilde{q} \}$ 
5:  $j = 1$ 
6:  $v_1 = \frac{[Q]_{:,1}}{\sqrt{|[\check{Q}]_{:,1}^* [Q]_{:,1}|}}$ ;  $w_1 = \frac{[\check{Q}]_{:,1}}{\text{sign}([\check{Q}]_{:,1}^* [Q]_{:,1}) \sqrt{|[\check{Q}]_{:,1}^* [Q]_{:,1}|}}$  ▷ Berechnung der ersten Basisvektoren
7: while  $j \leq \hat{k}$  ▷ Berechnung der folgenden Basisvektoren
8:     if  $j \leq \tilde{m}$ 
9:          $r_j = [Q]_{:,j}$  ▷ Folgender Basisvektor für  $V$ 
10:    else
11:         $r_j = P v_{j-\tilde{m}}$ 
12:    end if
13:    if  $j \leq \tilde{q}$ 
14:         $l_j = [\check{Q}]_{:,j}$  ▷ Folgender Basisvektor für  $W$ 
15:    else
16:         $l_j = \check{P} w_{j-\tilde{q}}$ 
17:    end if
18:    for  $i = 1 : j - 1$  ▷ Biorthogonalisierung
19:         $\delta_{i,j-1} = l_j^* v_i$ ;  $\epsilon_{i,j-1} = r_j^* w_i$ 
20:         $r_j = r_j - \epsilon_{i,j-1} v_i$ ;  $l_j = l_j - \delta_{i,j-1} w_i$ 
21:    end for
22:    if  $r_j^* l_j \geq \tau$ 
23:         $v_j = \frac{r_j}{\sqrt{|l_j^* r_j|}}$ ;  $w_j = \frac{l_j}{\text{sign}(l_j^* r_j) \sqrt{|l_j^* r_j|}}$ ;  $j = j + 1$  ▷ Normierung
24:    else
25:        if  $\|r_j\|_2 < \|l_j\|_2$  ▷ Deflation
26:            if  $j \leq \tilde{m}$ 
27:                 $Q = [ [Q]_{:,1:j-1} [Q]_{:,j+1:\tilde{m}} ]$  ▷ Löschen des  $j$ -ten Spaltenvektors der Matrix  $Q$ 
28:            end if
29:             $\tilde{m} = \tilde{m} - 1$  ▷ Spaltenanzahl der Matrix  $Q$  um 1 reduziert
30:            if  $\tilde{m} == 0$ 
31:                Abbruch der Schleife
32:            end if
33:        else
34:            if  $j \leq \tilde{q}$ 
35:                 $\check{Q} = [ [\check{Q}]_{:,1:j-1} [\check{Q}]_{:,j+1:\tilde{q}} ]$  ▷ Löschen des  $j$ -ten Spaltenvektors der Matrix  $\check{Q}$ 
36:            end if
37:             $\tilde{q} = \tilde{q} - 1$  ▷ Spaltenanzahl der Matrix  $\check{Q}$  um 1 reduziert
38:            if  $\tilde{q} == 0$ 
39:                Abbruch der Schleife
40:            end if
41:        end if
42:    end if
43: end while
44:  $V = [v_1, v_2, \dots, v_{\hat{k}}]$ ;  $W = [w_1, w_2, \dots, w_{\hat{k}}]$ 
45:  $W = (W^* V)^{-1} W^*$  ▷ Biorthogonalisierung

```

gen. Für SISO-Systeme werden in [81] entsprechende rationale Lanczos- und rationale Arnoldi-Verfahren für SISO-Systeme vorgeschlagen. Verfahren mit der Erweiterung für MIMO-Systeme ohne Deflation sind in [55, 136] zu finden. In [141] wird ein zweiseitiges rationales Arnoldi-Verfahren für MIMO-Systeme mit Deflation vorgestellt.

Allgemein gilt, dass der Arnoldi-Algorithmus einfacher zu implementieren und numerisch stabiler als der Lanczos-Algorithmus ist [81, 155]. Deshalb wird in dieser Arbeit für Verfahren zur Modellreduktion von MIMO-Systemen die Blockversion des rationalen Arnoldi-Algorithmus implementiert. Der Ablauf dieses Algorithmus orientiert sich an [141] und ist hier in Pseudocode als Algorithmus 5 wiedergegeben. Aufgrund von Deflation und/oder einer unterschiedlichen Anzahl

Algorithmus 5 Zweiseitiges rationales Arnoldi-Verfahren

Eingabe: Matrizen $\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}$; Vektor mit Stufen der Krylov-Unterräume \mathbf{s} ; Vektoren mit Stufen der Block-Krylov-Unterräume \mathbf{k}_b und \mathbf{k}_c ; Deflationstoleranz τ

Ausgabe: Basen \mathbf{V} und \mathbf{W}

```

1: function [  $\mathbf{V}, \mathbf{W}$  ] = Rationaler_Arnoldi( $\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{s}, \mathbf{k}_b, \mathbf{k}_c, \tau$ )
2:  $\mathbf{V} = []$ ,  $\mathbf{W} = []$  ▷ Initialisierungen
3:  $\hat{i} = \text{length}(\mathbf{s})$ 
4: for  $i = 1 : \hat{i}$ 
5:   Berechne die  $LU$ -Zerlegung:  $\mathbf{LU} = (\mathbf{A} - [\mathbf{s}]_i \mathbf{E})$ 
6:    $\mathbf{P} = \mathbf{U} \setminus (\mathbf{L} \setminus \mathbf{E})$ 
7:    $\mathbf{Q} = -\mathbf{U} \setminus (\mathbf{L} \setminus \mathbf{B})$ 
8:   Berechne  $[\mathbf{V}_i] = \text{Arnoldi}(\mathbf{P}, \mathbf{Q}, [\mathbf{k}_b]_i, \tau)$ ; Algorithmus 3 so modifiziert, dass  $\mathbf{V}_i$  orthogonal zu  $\mathbf{V}$  ist
9:    $\mathbf{P} = \mathbf{L}^* \setminus (\mathbf{U}^* \setminus \mathbf{E}^*)$ 
10:   $\mathbf{Q} = -\mathbf{L}^* \setminus (\mathbf{U}^* \setminus \mathbf{C}^*)$ 
11:  Berechne  $[\mathbf{W}_i] = \text{Arnoldi}(\mathbf{P}, \mathbf{Q}, [\mathbf{k}_c]_i, \tau)$ ; Algorithmus 3 so modifiziert, dass  $\mathbf{W}_i$  orthogonal zu  $\mathbf{W}$  ist
12:   $\mathbf{V} = [\mathbf{V} \ \mathbf{V}_i]$ ,  $\mathbf{W} = [\mathbf{W} \ \mathbf{W}_i]$ 
13:  if  $\text{Spaltenanzahl}(\mathbf{V}) \neq \text{Spaltenanzahl}(\mathbf{W})$ 
14:    Ergänzung von  $\mathbf{V}$  oder  $\mathbf{W}$ , um identische Anzahl an Basisvektoren zu erhalten
15:  end if
16: end for
```

an Eingängen und Ausgängen ist es möglich, dass die Dimensionen der Basen \mathbf{V} und \mathbf{W} nicht übereinstimmen. Die Basis mit niedrigerer Dimension muss deshalb um weitere Basisvektoren ergänzt werden. Da die LU -Zerlegung zur letzten Entwicklungsstelle noch verfügbar ist, wird in [140] vorgeschlagen weitere Elemente des Block-Krylov-Unterraumes dieser Entwicklungsstelle zu verwenden. Eine alternative Strategie besteht darin, die fehlenden Basisvektoren eines jeden Blockes \mathbf{V}_i bzw. \mathbf{W}_i zu einer Entwicklungsstelle durch zusätzlich generierte Spaltenvektoren zu ergänzen. Kann mit diesen Methoden der Unterraum aufgrund von Deflation nicht weiter vergrößert werden, so wird die Verwendung von zufällig generierten Vektoren als zusätzliche Basisvektoren empfohlen.

Die Dimension r der mit Hilfe dieses Algorithmus erzeugten Basen \mathbf{V} und \mathbf{W} wird, wenn während des Ablaufs des Algorithmus keine Deflation auftritt, durch die Anzahl \tilde{m} und der Anzahl \tilde{q} der linear unabhängigen Spalten der Eingangsmatrix \mathbf{B} bzw. der Ausgangsmatrix \mathbf{C} und der gewählten Stufe des Krylov-Unterraumes k mit $r = \max\{\tilde{m}, \tilde{q}\} \cdot k$ festgelegt.

Für die numerische Stabilität des Arnoldi-Verfahrens ist die Art der Orthogonalisierung entscheidend. In [152] wird dargelegt, dass sich die klassische Gram-Schmidt-Orthogonalisierung

als wenig geeignet erweist. In der Praxis zeigt sich, dass die Orthogonalität der berechneten Spaltenvektoren mit diesem Verfahren wegen numerischer Rundungsfehler sehr schnell verloren gehen kann. Eine numerisch stabilere Methode ist das modifizierte Gram-Schmidt-Verfahren mit Reorthogonalisierung, das hier in Algorithmus 4 zur Orthogonalisierung der Basisvektoren verwendet wird. In [140, 152] werden zur weiteren Effizienzsteigerung Kriterien vorgeschlagen, mit denen entschieden werden kann, ob eine Reorthogonalisierung für den aktuellen Basisvektor notwendig ist.

8.4.3.6 Erzielung reellwertiger Systeme

Werden die Verfahren mit komplexen Entwicklungsstellen ($s_i \in \mathbb{C}$ mit $\text{Im}(s_i) \neq 0$) verwendet, so ist eine ggf. gewünschte Reellwertigkeit der erzeugten Projektionsmatrizen im allgemeinen nicht gegeben. Diese Eigenschaft kann jedoch durch Anwendung geeigneter Methoden sichergestellt werden. In [151] wird eine Methode vorgeschlagen, die darauf beruht, dass neben den Entwicklungsstellen s_i , $i = 1, \dots, \hat{i}$ auch implizit die konjugiert komplexen Entwicklungsstellen \bar{s}_i , $i = 1, \dots, \hat{i}$ verwendet werden. In der Theorie kann damit eine reellwertige Projektionsmatrix \mathbf{V} generiert werden, jedoch zeigt sich in der praktischen Berechnung, dass die Reellwertigkeit der Projektionsmatrix, wegen numerischer Berechnungsfehler, nicht garantiert werden kann. Eine weitere Möglichkeit, eine reellwertige Projektionsmatrix \mathbf{V} zu erhalten, besteht in der Aufteilung der komplexen Projektionsmatrix $\mathbf{V} \in \mathbb{C}^{2n \times r}$ in ihren Real- und Imaginärteil

$$\mathbf{V}_{real} = [\text{Re}([\mathbf{V}]_{*1}) \cdots \text{Re}([\mathbf{V}]_{*r})] \quad \text{und} \quad \mathbf{V}_{imag} = [\text{Im}([\mathbf{V}]_{*1}) \cdots \text{Im}([\mathbf{V}]_{*r})] \quad (8.57)$$

und anschließender Reorthogonalisierung der Spaltenvektoren der Matrizen \mathbf{V}_{real} und \mathbf{V}_{imag} z. B. mit einer reduzierten \mathbf{QR} -Zerlegung

$$\mathbf{V} = \text{qr}([\mathbf{V}_{real} \ \mathbf{V}_{imag}]), \quad (8.58)$$

die nach der Berechnung von $2r$ orthogonalen Spalten der Matrix \mathbf{V} abbricht. Bei einem zweiseitigen Verfahren erhält man eine reellwertige Projektionsmatrix \mathbf{W} durch analoges Vorgehen. Bei Anwendung dieser Methode ist zu beachten, dass sich die Dimension des reduzierten Systems verdoppelt, ohne dass die Anzahl der Momente, die mit den Momenten der Übertragungsfunktion des Originalsystems übereinstimmen, erhöht wird.

8.4.4 Verfahren für Systeme zweiter Ordnung

Um die Grundidee der Verfahren für Systeme zweiter Ordnung basierend auf Krylov-Unterräumen zweiter Ordnung zu erläutern, wird das System zweiter Ordnung (3.1) in die verallgemeinerte Zustandsraumdarstellung (3.4) überführt. In dieser Darstellungsform wird analog zu den Krylov-Unterräumen aus Abschnitt 8.4.3 die Bestimmung geeigneter Basen für Krylov-Unterräume zweiter Ordnung anhand des Momentenabgleiches erläutert. Für die Transformation in die verallgemeinerte Zustandsraumdarstellung wird dabei die Darstellung (3.6) mit $\mathbf{N} = -\mathbf{K}$ verwendet.

Für den Eingangs-Krylov-Unterraum \mathcal{K}_k wie in (8.44) erhält man dann

$$\begin{aligned} \mathcal{K}_k(\mathbf{A}^{-1} \mathbf{E}, -\mathbf{A}^{-1} \mathbf{b}) = \\ \mathcal{K}_k \left(\begin{bmatrix} -\mathbf{K} & \mathbf{0} \\ \mathbf{0} & -\mathbf{K} \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{D} & \mathbf{M} \\ -\mathbf{K} & \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K} & \mathbf{0} \\ \mathbf{0} & \mathbf{K} \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix} \right) = \\ \mathcal{K}_k \left(\begin{bmatrix} -\mathbf{K}^{-1} \mathbf{D} & -\mathbf{K}^{-1} \mathbf{M} \\ \mathbf{I} & \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K}^{-1} \mathbf{f} \\ \mathbf{0} \end{bmatrix} \right) \end{aligned} \quad (8.59)$$

bzw.

$$\begin{aligned} \mathcal{K}_k(\mathbf{A}^{-1} \mathbf{E}, -\mathbf{A}^{-1} \mathbf{B}) = \\ \mathcal{K}_k \left(\begin{bmatrix} -\mathbf{K} & \mathbf{0} \\ \mathbf{0} & -\mathbf{K} \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{D} & \mathbf{M} \\ -\mathbf{K} & \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K} & \mathbf{0} \\ \mathbf{0} & \mathbf{K} \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{F} \\ \mathbf{0} \end{bmatrix} \right) = \\ \mathcal{K}_k \left(\begin{bmatrix} -\mathbf{K}^{-1} \mathbf{D} & -\mathbf{K}^{-1} \mathbf{M} \\ \mathbf{I} & \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K}^{-1} \mathbf{F} \\ \mathbf{0} \end{bmatrix} \right). \end{aligned} \quad (8.60)$$

Die Krylov-Unterräume (8.59) und (8.60) besitzen die allgemeine Struktur

$$\mathcal{K}_k \left(\begin{bmatrix} \mathbf{P}_1 & \mathbf{P}_2 \\ \mathbf{I} & \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{Q} \\ \mathbf{0} \end{bmatrix} \right), \quad (8.61)$$

wodurch sich eine Krylov-Sequenz der Form

$$\begin{bmatrix} \mathbf{X}_0 & \mathbf{X}_1 & \cdots & \mathbf{X}_{k-1} \\ \mathbf{0} & \mathbf{X}_0 & \cdots & \mathbf{X}_{k-2} \end{bmatrix} = \begin{bmatrix} \mathbf{Z}_o \\ \mathbf{Z}_u \end{bmatrix} \quad (8.62)$$

mit

$$\begin{aligned} \mathbf{X}_0 &= \mathbf{K}^{-1} \mathbf{F} = \mathbf{Q}, \\ \mathbf{X}_1 &= -\mathbf{K}^{-1} \mathbf{D} \mathbf{X}_0 = \mathbf{P}_1 \mathbf{X}_0, \\ \mathbf{X}_i &= -\mathbf{K}^{-1} \mathbf{D} \mathbf{X}_{i-1} - \mathbf{K}^{-1} \mathbf{M} \mathbf{X}_{i-2} = \mathbf{P}_1 \mathbf{X}_{i-1} + \mathbf{P}_2 \mathbf{X}_{i-2}, \quad \text{für } i = 2, \dots, k-1 \end{aligned} \quad (8.63)$$

ergibt. Aus (8.62) ist ersichtlich, dass

$$\text{colspan}(\mathbf{Z}_u) \subseteq \text{colspan}(\mathbf{Z}_o). \quad (8.64)$$

Durch \mathbf{Z}_o wird somit bereits die gesamte Krylov-Sequenz festgelegt.

Analog zu Systemen erster Ordnung lässt sich auch für Systeme zweiter Ordnung der Ausgangs-Krylov-Unterraum \mathcal{K}_k wie in (8.46) bestimmen. Für Systeme zweiter Ordnung der Form (3.1) weisen die Matrix \mathbf{P} und \mathbf{q} bzw. \mathbf{Q} in (2.48) bzw. (2.50) jedoch eine kompliziertere Struktur auf. Geht man wieder von der Transformation in die verallgemeinerte Zustandsraumdarstellung

der Form (3.6) aus, dann gilt zunächst nur

$$\begin{aligned} \mathcal{K}_k(\mathbf{A}^{-*} \mathbf{E}^*, -\mathbf{A}^{-*} \mathbf{c}^*) = \\ \mathcal{K}_k \left(\begin{bmatrix} -\mathbf{K}^* & \mathbf{0} \\ \mathbf{0} & \mathbf{N}^* \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{D}^* & \mathbf{N}^* \\ \mathbf{M}^* & \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K}^* & \mathbf{0} \\ \mathbf{0} & -\mathbf{N}^* \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{c}_p^* \\ \mathbf{c}_v^* \end{bmatrix} \right) = \\ \mathcal{K}_k \left(\begin{bmatrix} -\mathbf{K}^{-*} \mathbf{D}^* & -\mathbf{K}^{-*} \mathbf{N}^* \\ \mathbf{N}^{-*} \mathbf{M}^* & \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K}^{-*} \mathbf{c}_p^* \\ -\mathbf{N}^{-*} \mathbf{c}_v^* \end{bmatrix} \right) \end{aligned} \quad (8.65)$$

bzw.

$$\begin{aligned} \mathcal{K}_k(\mathbf{A}^{-*} \mathbf{E}^*, -\mathbf{A}^{-*} \mathbf{C}^*) = \\ \mathcal{K}_k \left(\begin{bmatrix} -\mathbf{K}^* & \mathbf{0} \\ \mathbf{0} & \mathbf{N}^* \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{D}^* & \mathbf{N}^* \\ \mathbf{M}^* & \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K}^* & \mathbf{0} \\ \mathbf{0} & -\mathbf{N}^* \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{C}_p^* \\ \mathbf{C}_v^* \end{bmatrix} \right) = \\ \mathcal{K}_k \left(\begin{bmatrix} -\mathbf{K}^{-*} \mathbf{D}^* & -\mathbf{K}^{-*} \mathbf{N}^* \\ \mathbf{N}^{-*} \mathbf{M}^* & \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K}^{-*} \mathbf{C}_p^* \\ -\mathbf{N}^{-*} \mathbf{C}_v^* \end{bmatrix} \right). \end{aligned} \quad (8.66)$$

Setzt man jedoch voraus, dass die Matrix \mathbf{M} invertierbar ist und setzt $\mathbf{N} = \mathbf{M}$, besitzen die Krylov-Unterräume (8.65) und (8.66) die Struktur

$$\mathcal{K}_k \left(\begin{bmatrix} \mathbf{P}_1 & \mathbf{P}_2 \\ \mathbf{I} & \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \end{bmatrix} \right), \quad (8.67)$$

wodurch sich eine zu (8.62) ähnliche Krylov-Sequenz der Form

$$\begin{bmatrix} \mathbf{X}_0 & \mathbf{X}_1 & \cdots & \mathbf{X}_{k-1} \\ \mathbf{X}_{-1} & \mathbf{X}_0 & \cdots & \mathbf{X}_{k-2} \end{bmatrix} = \begin{bmatrix} \mathbf{Z}_o \\ \mathbf{Z}_u \end{bmatrix} \quad (8.68)$$

mit

$$\begin{aligned} \mathbf{X}_{-1} &= -\mathbf{M}^{-*} \mathbf{C}_v^* = \mathbf{Q}_2, \\ \mathbf{X}_0 &= \mathbf{K}^{-*} \mathbf{C}_p^* = \mathbf{Q}_1, \\ \mathbf{X}_i &= -\mathbf{K}^{-*} \mathbf{D}^* \mathbf{X}_{i-1} - \mathbf{K}^{-*} \mathbf{M}^* \mathbf{X}_{i-2} = \mathbf{P}_1 \mathbf{X}_{i-1} + \mathbf{P}_2 \mathbf{X}_{i-2}, \quad \text{für } i = 1, \dots, k-1 \end{aligned} \quad (8.69)$$

ergibt.

Betrachtet man ein System zweiter Ordnung der Form (3.1) mit $\mathbf{C}_v = \mathbf{0}$, weisen die Krylov-Unterräume (8.67) die selbe Struktur (8.61), wie die Eingangs-Krylov-Unterräume (8.59) bzw. (8.60) auf, wodurch sich eine Krylov-Sequenz der Form (8.62) mit

$$\begin{aligned} \mathbf{X}_0 &= \mathbf{K}^{-*} \mathbf{C}_p = \mathbf{Q}, \\ \mathbf{X}_1 &= -\mathbf{K}^{-*} (\mathbf{D}^* \mathbf{X}_0) = \mathbf{P}_1 \mathbf{X}_0, \\ \mathbf{X}_i &= -\mathbf{K}^{-*} \mathbf{D}^* \mathbf{X}_{i-1} - \mathbf{K}^{-*} \mathbf{M}^* \mathbf{X}_{i-2} = \mathbf{P}_1 \mathbf{X}_{i-1} + \mathbf{P}_2 \mathbf{X}_{i-2}, \quad \text{für } i = 2, \dots, k-1 \end{aligned} \quad (8.70)$$

ergibt.

Mit der Definition des Krylov-Unterraumes zweiter Ordnung (2.52) und der Rekursion (8.63) bzw. (8.70) erhält man den Krylov-Unterraum zweiter Ordnung der Form

$$\mathcal{G}_k(\mathbf{P}_1, \mathbf{P}_2, \mathbf{Q}) = \text{colspan}\{\mathbf{X}_0, \mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_{k-1}\} = \text{colspan}(\mathbf{Z}_o). \quad (8.71)$$

Die Basis dieses Krylov-Unterraum zweiter Ordnung (8.71) spannt somit den gleichen Raum auf, wie die Basis der oberen Hälfte des Krylov-Unterraumes (8.61).

Es folgt, dass eine Matrix

$$\tilde{\mathbf{Z}} = \begin{bmatrix} \mathbf{Z}_o & \mathbf{0} \\ \mathbf{0} & \mathbf{Z}_o \end{bmatrix} \quad (8.72)$$

mit der Eigenschaft

$$\mathcal{K}_k \left(\begin{bmatrix} \mathbf{P}_1 & \mathbf{P}_2 \\ \mathbf{I} & \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{Q} \\ \mathbf{0} \end{bmatrix} \right) \subseteq \text{colspan}(\tilde{\mathbf{Z}}) \quad (8.73)$$

existiert.

Mit geeigneter Wahl von \mathbf{P}_1 , \mathbf{P}_2 und \mathbf{Q} in (8.71) kann demnach die Projektionsmatrix

$$\tilde{\mathbf{V}} = \begin{bmatrix} \mathbf{V}_o & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_o \end{bmatrix}, \quad \text{bzw.} \quad \tilde{\mathbf{W}} = \begin{bmatrix} \mathbf{W}_o & \mathbf{0} \\ \mathbf{0} & \mathbf{W}_o \end{bmatrix} \quad (8.74)$$

definiert werden, wobei die Matrizen \mathbf{V}_o und \mathbf{W}_o Basen des Eingangs- bzw. des Ausgangs-Krylov-Unterraumes zweiter Ordnung (8.71) darstellen.

Damit lässt sich die Reduktion von Systemen zweiter Ordnung mit Hilfe von zwei unterschiedlichen Methoden durchführen:

- a) Transformation des Systems zweiter Ordnung in die verallgemeinerte Zustandsraumdarstellung (3.4) und anschliessender Projektion mit Matrizen $\tilde{\mathbf{V}}$ und $\tilde{\mathbf{W}}$ der Form (8.74).
- b) Direkte Projektion des Systems zweiter Ordnung mit Hilfe der Matrizen $\mathbf{V} = \mathbf{V}_o$ und $\mathbf{W} = \mathbf{W}_o$, die Basen eines Eingangs- bzw. Ausgangs-Krylov-Unterräumes zweiter Ordnung darstellen.

Dabei ist die direkte Anwendung der Projektion auf das System zweiter Ordnung unter Verwendung der Projektionsmatrizen \mathbf{V}_o und \mathbf{W}_o äquivalent zu der Projektion des äquivalenten Systems erster Ordnung mit den blockdiagonalen Matrizen $\tilde{\mathbf{V}}$ und $\tilde{\mathbf{W}}$, siehe (5.19) in Abschnitt 5.2. Die Eigenschaften des Momentenabgleiches des reduzierten Systems folgen aufgrund von (8.73) dann aus den Approximationseigenschaften von $\tilde{\mathbf{V}}$ und $\tilde{\mathbf{W}}$.

8.4.4.1 Verfahren mit $s_0 = 0$

Wählt man die Projektionsmatrizen $\mathbf{V}, \mathbf{W} \in \mathbb{K}^{n \times r}$ für das Projektionsverfahren (5.16) aus Abschnitt 5.2 mit

$$\begin{aligned} \mathbf{P}_1 &= -\mathbf{K}^{-1} \mathbf{D}, \\ \mathbf{P}_2 &= -\mathbf{K}^{-1} \mathbf{M} \quad \text{und} \\ \mathbf{q} &= \mathbf{K}^{-1} \mathbf{f} \quad \text{bzw.} \quad \mathbf{Q} = \mathbf{K}^{-1} \mathbf{F} \end{aligned} \quad (8.75)$$

in (2.52) so, dass gilt

$$\begin{aligned} \mathcal{G}_k(-K^{-1}D, -K^{-1}M, K^{-1}f) &\subseteq \mathcal{V} = \text{colspan}(\mathbf{V}) \quad \text{bzw.} \\ \mathcal{G}_k(-K^{-1}D, -K^{-1}M, K^{-1}F) &\subseteq \mathcal{V} = \text{colspan}(\mathbf{V}), \\ \mathbf{W} &= \mathbf{V} \end{aligned} \quad (8.76)$$

und die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist, dann erhält man ein einseitiges Verfahren, das auf der Berechnung des sogenannten (SISO-) Eingangs-Krylov-Unterraumes zweiter Ordnung bzw. des sogenannten (MIMO-) Eingangs-Krylov-Unterraum zweiter Ordnung basiert.

Satz 8.9 ((Eingangs-) Momentenabgleich an der Stelle $s_0 = 0$) *Sei ein System zweiter Ordnung der Form (3.1) gegeben. Bilden die Spaltenvektoren einer Projektionsmatrix $\mathbf{V} \in \mathbb{K}^{n \times r}$ eine Basis des Krylov-Unterraumes $\mathcal{G}_k(\mathbf{P}_1, \mathbf{P}_2, \mathbf{q})$ bzw. des Block-Krylov-Unterraumes $\mathcal{G}_k(\mathbf{P}_1, \mathbf{P}_2, \mathbf{Q})$, mit \mathbf{P}_1 , \mathbf{P}_2 und \mathbf{q} bzw. \mathbf{Q} wie in (8.75) und wird die Projektionsmatrix $\mathbf{W} \in \mathbb{K}^{n \times r}$ so gewählt, dass die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist, dann stimmen mindestens die ersten $\lfloor r/m \rfloor$ Momente der Übertragungsfunktion des mit der Projektion $\mathbf{\Pi} = \mathbf{V} \mathbf{W}^*$ erzeugten reduzierten Systems und des originalen Systems an der Entwicklungsstelle $s_0 = 0$ überein.*

BEWEIS: Siehe z. B. [155, 156]. □

Alternativ kann für Systeme zweiter Ordnung der Form (3.1) mit $\mathbf{C}_v = \mathbf{0}$ auch der Ausgangs-Krylov-Unterraum zweiter Ordnung zum Bestimmen geeigneter Spaltenvektoren der Projektionsmatrix \mathbf{W} verwendet werden. Dazu wählt man die Projektionsmatrizen $\mathbf{V}, \mathbf{W} \in \mathbb{K}^{n \times r}$ für das Projektionsverfahren (5.16) aus Abschnitt 5.2 mit

$$\begin{aligned} \mathbf{P}_1 &= -K^{-*} D^*, \\ \mathbf{P}_2 &= -K^{-*} M^* \quad \text{und} \\ \mathbf{q} &= K^{-*} \mathbf{c}_p^* \quad \text{bzw.} \quad \mathbf{Q} = K^{-*} \mathbf{C}_p^* \end{aligned} \quad (8.77)$$

in (2.52) so, dass gilt

$$\begin{aligned} \mathcal{G}_k(-K^{-*}D, -K^{-*}M, K^{-*}\mathbf{c}_p^*) &\subseteq \mathcal{W} = \text{colspan}(\mathbf{W}) \quad \text{bzw.} \\ \mathcal{G}_k(-K^{-*}D, -K^{-*}M, K^{-*}\mathbf{C}_p^*) &\subseteq \mathcal{W} = \text{colspan}(\mathbf{W}), \\ \mathbf{V} &= \mathbf{W} \end{aligned} \quad (8.78)$$

und die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist, dann erhält man ein einseitiges Verfahren, das auf der Berechnung des sogenannten (SISO-) Ausgangs-Krylov-Unterraumes zweiter Ordnung bzw. des sogenannten (MIMO-) Ausgangs-Krylov-Unterraumes zweiter Ordnung basiert.

Für den Momentenabgleich reduzierter Systeme, die mit diesen einseitigen Verfahren erzeugt werden, gilt die Aussage des folgenden Satzes, siehe [155, 156].

Satz 8.10 ((Ausgangs-) Momentenabgleich an der Stelle $s_0 = 0$) *Sei ein System zweiter Ordnung der Form (3.1) mit $\mathbf{C}_v = 0$ gegeben. Bilden die Spaltenvektoren einer Projektionsmatrix $\mathbf{V} \in \mathbb{K}^{n \times r}$ eine Basis des Krylov-Unterraumes $\mathcal{G}_k(\mathbf{P}_1, \mathbf{P}_2, \mathbf{q})$ bzw. des Block-Krylov-Unterraumes $\mathcal{G}_k(\mathbf{P}_1, \mathbf{P}_2, \mathbf{Q})$, mit \mathbf{P}_1 , \mathbf{P}_2 und \mathbf{q} bzw. \mathbf{Q} wie in (8.77) und wird die Projektionsmatrix $\mathbf{W} \in \mathbb{K}^{n \times r}$ so gewählt, dass die Matrix $\mathbf{W}^* \mathbf{A} \mathbf{V}$ regulär ist, dann stimmen mindestens die*

ersten die ersten $\lfloor r/q \rfloor$ Momente der Übertragungsfunktion des mit der Projektion $\Pi = V W^*$ erzeugten reduzierten Systems und des originalen Systems an der Entwicklungsstelle $s_0 = 0$ überein.

BEWEIS: Siehe z. B. [155]. □

Wählt man für das Projektionsverfahren die Projektionsmatrix V wie in (8.76) und die Projektionsmatrix W wie in (8.78), so erhält man zweiseitige Verfahren. Der folgende Satz macht Aussagen bezüglich des Momentenabgleiches der reduzierten Systeme, die mit Hilfe dieser Verfahren erzeugt werden, siehe [155].

Satz 8.11 ((Zweiseitiger) Momentenabgleich an der Stelle $s_0 = 0$) Sei ein System zweiter Ordnung der Form (3.1) mit $C_v = 0$ gegeben. Wird die Projektionsmatrix $V \in \mathbb{K}^{n \times r}$ wie in Satz 8.9 und die Projektionsmatrix $W \in \mathbb{K}^{n \times r}$ wie in Satz 8.10 mit $W^* K V$ regulär gebildet, dann stimmen mindestens die ersten $\lfloor r/m \rfloor + \lfloor r/q \rfloor$ Momente der Übertragungsfunktion des mit der Projektion $\Pi = V W^*$ erzeugten reduzierten Systems und des originalen Systems an der Entwicklungsstelle $s_0 = 0$ überein.

BEWEIS: Siehe [155]. □

8.4.4.2 Verfahren mit $s_i \in \mathbb{C}$

In Analogie zu den Systemen erster Ordnung kann die Methode des Momentenabgleichs auch für Systeme zweiter Ordnung auf beliebige Entwicklungsstellen $s_i \in \mathbb{C}$ erweitert werden.

Dazu betrachtet man die geshiftete Übertragungsfunktion (3.74) des Systems zweiter Ordnung der Form (3.1):

$$H(s) = (\tilde{C}_{p,i} + \tilde{s} C_v) \cdot (\tilde{s}^2 M + \tilde{s} \tilde{D}_i + \tilde{K}_i)^{-1} \cdot F, \quad (8.79)$$

wobei

$$\begin{aligned} \tilde{s} &= (s - s_i), \\ \tilde{D}_i &= 2 s_i M + D, \\ \tilde{K}_i &= s_i^2 M + s_i D + K \quad \text{und} \\ \tilde{C}_{p,i} &= C_p + s_i C_v \end{aligned} \quad (8.80)$$

gilt [15, 122, 155].

Damit ist die Potenzreihenentwicklung der Übertragungsfunktion $H(s)$ an der Entwicklungsstelle s_i demnach äquivalent zu einer Entwicklung der geshifteten Übertragungsfunktion (8.79) am Nullpunkt. Durch Ersetzen der Systemmatrizen D, K, C_p in (8.59), (8.60), (8.65) und (8.66) durch die Matrizen $\tilde{D}_i, \tilde{K}_i, \tilde{C}_{p,i}$ kann somit Momentenabgleich an einer beliebigen Entwicklungsstelle $s_i \in \mathbb{C}$ realisiert werden.

8.4.4.3 Verfahren mit mehr als einer Entwicklungsstelle

Bei Verfahren, die den Eingangs-Krylov-Unterraum verwenden, werden die Projektionsmatrizen V und W für das Projektionsverfahren (5.16) aus Abschnitt 5.2 für $i = 1, \dots, \hat{i}$ mit

$$\begin{aligned} P_1 &= P_{1,i} = -\tilde{K}_i^{-1} \tilde{D}_i, \\ P_2 &= P_{2,i} = -\tilde{K}_i^{-1} M, \quad \text{und} \\ q &= q_i = \tilde{K}_i^{-1} f \quad \text{bzw.} \quad Q = Q_i = \tilde{K}_i^{-1} F \end{aligned} \quad (8.81)$$

in (2.52) so gewählt, dass gilt

$$\begin{aligned} \bigcup_{i=1}^{\hat{i}} \mathcal{G}_{k(i)}(-\tilde{K}_i^{-1} \tilde{D}_i, -\tilde{K}_i^{-1} M, \tilde{K}_i^{-1} f) &\subseteq \mathcal{V} = \text{colspan}(V) \quad \text{bzw.} \\ \bigcup_{i=1}^{\hat{i}} \mathcal{G}_{k(i)}(-\tilde{K}_i^{-1} \tilde{D}_i, -\tilde{K}_i^{-1} M, \tilde{K}_i^{-1} F) &\subseteq \mathcal{V} = \text{colspan}(V), \end{aligned} \quad (8.82)$$

$$W = V$$

und die Matrix $W^* A V$ regulär ist.

Satz 8.12 ((Eingangs-) Momentabgleich an einer Stelle $s_i \in \mathbb{C}$) Sei ein System zweiter Ordnung der Form (3.1) gegeben. Bilden die Spaltenvektoren einer Projektionsmatrix $V \in \mathbb{K}^{n \times r}$ eine Basis des Krylov-Unterraumes $\bigcup_{i=1}^{\hat{i}} \mathcal{G}_{k(i)}(P_{1,i}, P_{2,i}, q_i)$ bzw. des Block-Krylov-Unterraumes $\bigcup_{i=1}^{\hat{i}} \mathcal{G}_{k(i)}(P_{1,i}, P_{2,i}, Q_i)$ mit $P_{1,i}$, $P_{2,i}$ und q_i bzw. Q_i wie in (8.81) und wird die Projektionsmatrix $W \in \mathbb{K}^{n \times r}$ so gewählt, dass die Matrix $W^* A V$ regulär ist, dann stimmen mindestens die ersten $\lfloor r/(m \cdot \hat{i}) \rfloor$ Momente der Übertragungsfunktion des mit der Projektion $\Pi = V W^*$ erzeugten reduzierten Systems und des originalen Systems an der Entwicklungsstelle $s_i \in \mathbb{C}$, $i = 1, 2, \dots, \hat{i}$ überein.

BEWEIS: Siehe z. B. [155]. □

Bei Verfahren, die den Ausgangs-Krylov-Unterraum verwenden, werden die Projektionsmatrizen V und W für das Projektionsverfahren (5.16) aus Abschnitt 5.2 für $i = 1, \dots, \hat{i}$ mit

$$\begin{aligned} P_1 &= P_{1,i} = -\tilde{K}_i^{-*} \tilde{D}_i^*, \\ P_2 &= P_{2,i} = -\tilde{K}_i^{-*} M^*, \quad \text{und} \\ q &= q_i = \tilde{K}_i^{-*} c_p^* \quad \text{bzw.} \quad Q = Q_i = \tilde{K}_i^{-*} C_p^* \end{aligned} \quad (8.83)$$

in (2.52) so gewählt, dass gilt

$$\begin{aligned} \bigcup_{i=1}^{\hat{i}} \mathcal{G}_{k(i)}(-\tilde{K}_i^{-*} \tilde{D}_i^*, -\tilde{K}_i^{-*} M^*, \tilde{K}_i^{-*} c_p^*) &\subseteq \mathcal{W} = \text{colspan}(W) \quad \text{bzw.} \\ \bigcup_{i=1}^{\hat{i}} \mathcal{G}_{k(i)}(-\tilde{K}_i^{-*} \tilde{D}_i^*, -\tilde{K}_i^{-*} M^*, \tilde{K}_i^{-*} C_p^*) &\subseteq \mathcal{W} = \text{colspan}(W), \end{aligned} \quad (8.84)$$

$$V = W$$

und die Matrix $W^* A V$ regulär ist.

Satz 8.13 ((Ausgangs-) Momentenabgleich an einer Stelle $s_i \in \mathbb{C}$) Sei ein System zweiter Ordnung der Form (3.1) mit $C_v = \mathbf{0}$ gegeben. Bilden die Spaltenvektoren einer Projektionsmatrix $V \in \mathbb{K}^{n \times r}$ eine Basis des Krylov-Unterraumes $\bigcup_{i=1}^{\hat{i}} \mathcal{G}_{k(i)}(P_{1,i}, P_{2,i}, q_i)$ bzw. des Block-Krylov-Unterraumes $\bigcup_{i=1}^{\hat{i}} \mathcal{G}_{k(i)}(P_{1,i}, P_{2,i}, Q_i)$ mit $P_{1,i}$, $P_{2,i}$ und q_i bzw. Q_i wie in (8.83) und wird die Projektionsmatrix $W \in \mathbb{K}^{n \times r}$ so gewählt, dass die Matrix $W^* A V$ regulär ist, dann stimmen mindestens die ersten $\lfloor r/(q \cdot \hat{i}) \rfloor$ Momente der Übertragungsfunktion des mit der Projektion $\Pi = V W^*$ erzeugten reduzierten Systems und des originalen Systems an der Entwicklungsstelle $s_i \in \mathbb{C}$, $i = 1, 2, \dots, \hat{i}$ überein.

BEWEIS: Siehe z. B. [155]. □

Wählt man für das Projektionsverfahren die Projektionsmatrizen V wie in (8.82) und die Projektionsmatrizen W wie in (8.84), so erhält man zweiseitige Verfahren mit mehreren Entwicklungsstellen $s_i \in \mathbb{C}$, $i = 1, 2, \dots, \hat{i}$. Der folgende Satz macht Aussagen bezüglich des Momentenabgleiches der reduzierten Systeme, die mit Hilfe dieser Verfahren erzeugt werden.

Satz 8.14 ((Zweiseitiger) Momentenabgleich an einer Stelle $s_i \in \mathbb{C}$) Sei ein System zweiter Ordnung der Form (3.1) mit $C_v = \mathbf{0}$ gegeben. Wird die Projektionsmatrix $V \in \mathbb{K}^{n \times r}$ wie in Satz 8.12 und die Projektionsmatrix $W \in \mathbb{K}^{n \times r}$ wie in Satz 8.13 mit $W^* K V$ regulär gebildet, dann stimmen mindestens die ersten $\lfloor r/(m \cdot \hat{i}) \rfloor + \lfloor r/(q \cdot \hat{i}) \rfloor$ Momente der Übertragungsfunktion des mit der Projektion $\Pi = V W^*$ erzeugten reduzierten Systems und des originalen Systems an der Entwicklungsstelle $s_i \in \mathbb{C}$, $i = 1, 2, \dots, \hat{i}$ überein.

BEWEIS: Siehe [155]. □

Anmerkung 8.2 Für den wichtigen Sonderfall, dass ein reelles System mit symmetrischen Systemmatrizen M, D, K und mit $C_v = \mathbf{0}$, $C_p = F^*$ vorliegt und $\text{Im}(s_i) = 0$ ist, entspricht der Ausgangs-Krylov-Unterraum \mathcal{W} dem Eingangs-Krylov-Unterraum \mathcal{V} . Damit kann anstelle einer schiefen Projektion eine orthogonale Projektion mit $W = V$ und $V^* V = I$ verwendet werden, um mindestens die ersten $2 \cdot \lfloor r/(m \cdot \hat{i}) \rfloor$ Momente an den Entwicklungsstellen $s_i \in \mathbb{C}$, $i = 1, 2, \dots, \hat{i}$ zur Übereinstimmung zu bringen. Auch für den Fall $\text{Im}(s_i) \neq 0$ ist zum Erreichen dieses Zieles eine orthogonale Projektion anwendbar, indem für den Ausgangs-Krylov-Unterraum \mathcal{W} die konjugiert komplexen Entwicklungsstellen \bar{s}_i verwendet werden.

8.4.4.4 Numerische Aspekte

Um die für die Verfahren zur Modellreduktion von Systemen zweiter Ordnung benötigten Basen der Krylov-Unterräume zweiter Ordnung numerisch stabil zu berechnen, sind spezielle Methoden entwickelt worden. Als Beispiele solcher Methoden seien hier das Second-Order-Arnoldi-Verfahren (SOAR-Verfahren) nach [15, 155] und das Second-Order-Lanczos-Verfahren nach [155] genannt. Diese Verfahren, die auf [173] zurückgehen, erzeugen orthogonale Basen für Krylov-Unterräume zweiter Ordnung der Struktur (8.71), indem sie die spezielle Struktur des Krylov-Unterraumes (8.61) ausnutzen, um die benötigte orthonormale Basis V (und W) zu erzeugen. Bei diesen Verfahren ist die Orthogonalisierung der Basisvektoren des klassischen Arnoldi- oder Lanczos-Verfahrens so modifiziert, dass anstelle des gesamten Blocks der neu erzeugten Basisvek-

toren $[\mathbf{V}_o^T, \mathbf{V}_u^T]^T$ (und $[\mathbf{W}_o^T, \mathbf{W}_u^T]^T$) lediglich die obere Hälfte \mathbf{V}_o (und \mathbf{W}_o) die gewünschten Orthogonalisierungsbedingungen erfüllt. Der Algorithmus des Second Order Arnoldi-Verfahrens ist als Algorithmus 6 und der Algorithmus des Second Order Lanczos-Verfahrens ist als Algorithmus 7 in Pseudocode wiedergegeben. Bei diesen Verfahren bringt der Orthogonalisierungsprozess

Algorithmus 6 Second Order Arnoldi-Verfahren (SOAR) nach [15, 155]

Eingabe: Matrizen $\mathbf{P}_1, \mathbf{P}_2, \mathbf{Q}$; Stufe des Block-Krylov-Unterraumes k ; Deflationstoleranz τ

Ausgabe: Basis \mathbf{V}

```

1: function  $\mathbf{V} = \text{SOAR}(\mathbf{P}_1, \mathbf{P}_2, \mathbf{Q}, k, \tau)$ 
2: Entfernung aller linear abhängigen Spaltenvektoren der Matrix  $\mathbf{Q}$ 
3:  $\mathbf{V}_o = []$ ;  $\mathbf{V}_u = []$ 
4: for  $j = 1 : k$ 
5:     if  $(j == 1)$ 
6:          $\mathbf{R}_o = \mathbf{Q}$ ;  $\mathbf{R}_u = \mathbf{0}$  ▷ Startblöcke
7:     else if  $(j == 2)$ 
8:          $\mathbf{R}_u = \mathbf{R}_o$ ;  $\mathbf{R}_o = \mathbf{P}_1 \mathbf{R}_o$  ▷ Berechnung weiterer Blöcke
9:     else
10:         $\mathbf{R}_{old} = \mathbf{R}_o$  ▷ Berechnung weiterer Blöcke
11:         $\mathbf{R}_o = \mathbf{P}_1 \mathbf{R}_o + \mathbf{P}_2 \mathbf{R}_u$ ;  $\mathbf{R}_u = \mathbf{R}_{old}$ 
12:    end if
13:    for  $i = 1 : \text{Spaltenanzahl}(\mathbf{R}_o)$ 
14:        for  $t = 1 : \text{Spaltenanzahl}(\mathbf{V}_o)$ 
15:             $h = [\mathbf{R}_o]_{:,i}^* [\mathbf{V}_o]_{:,t}$  ▷ Skalarprodukt
16:             $[\mathbf{R}_o]_{:,i} = [\mathbf{R}_o]_{:,i} - h [\mathbf{V}_o]_{:,t}$ ;  $[\mathbf{R}_u]_{:,i} = [\mathbf{R}_u]_{:,i} - h [\mathbf{V}_u]_{:,t}$ 
17:        end for
18:         $h_o = \|\mathbf{R}_o\|_2$ ;  $h_u = \|\mathbf{R}_u\|_2$ 
19:        if  $(h_o > \tau)$  ▷ Deflation des Vektors?
20:             $\mathbf{v}_o = \frac{1}{h_o} [\mathbf{R}_o]_{:,i}$ ;  $\mathbf{v}_u = \frac{1}{h_o} [\mathbf{R}_u]_{:,i}$  ▷ Normierungen
21:             $\mathbf{V}_o = [\mathbf{V}_o \mathbf{v}_o]$ ;  $\mathbf{V}_u = [\mathbf{V}_u \mathbf{v}_u]$  ▷ Hinzufügen des Vektors
22:        else if  $(h_u > \tau)$ 
23:             $\mathbf{v}_o = \mathbf{0}$ ;  $\mathbf{v}_u = [\mathbf{R}_u]_{:,i}$ 
24:             $\mathbf{V}_o = [\mathbf{V}_o \mathbf{v}_o]$ ;  $\mathbf{V}_u = [\mathbf{V}_u \mathbf{v}_u]$  ▷ Hinzufügen des Vektors
25:        else
26:            Deflation
27:        end if
28:    end for
29: end for
30: Löschen von Nullspalten in  $\mathbf{V}_o$ 
31:  $\mathbf{V} = \mathbf{V}_o$ 
    
```

Schwierigkeiten bezüglich des Deflationskriteriums mit sich. Das Kriterium $h_o < \tau$ beim SOAR bzw. $\delta_o < \tau$ und $\epsilon_o < \tau$ beim Second Order Lanczos-Verfahren ist als alleiniges Kriterium für eine Deflation geeignet. Der Grund dafür liegt darin, dass bei Betrachtung von $h_o < \tau$ bzw. $\delta_o < \tau$ und $\epsilon_o < \tau$ ausschließlich eine Aussage bezüglich des oberen Teils der zu betrachtenen Krylov-Sequenz (8.62) gemacht werden kann. Als Deflationskriterium sollte aber die lineare Abhängigkeit der gesamten Sequenz dienen, siehe dazu [15, 109]. In den Algorithmen 6 und 7 wird deshalb zusätzlich das Kriterium h_u bzw. $\delta_u < \tau$ und $\epsilon_u < \tau$ verwendet, um ein geeignetes Deflationskriterium zu erhalten. Das in den hier dargestellten Algorithmen verwendete Deflationskriterium entspricht dem Vorgehen aus [155].

Algorithmus 7 Second Order Lanczos-Verfahren nach [155]**Eingabe:** Matrizen $P_1, P_2, \check{P}_1, \check{P}_2, Q, \check{Q}$; Stufe des Block-Krylov-Unterraumes k ; Deflationstoleranz τ **Ausgabe:** Basen V und W

```

1: function [  $V, W$  ] = Lanczos_2o( $P_1, P_2, \check{P}_1, \check{P}_2, Q, \check{Q}, k, \tau$ )
2: Entfernung aller linear abhängigen Spaltenvektoren der Matrizen  $Q$  und  $\check{Q}$ 
3:  $V_o = [ ]$ ;  $V_u = [ ]$ ;  $W_o = [ ]$ ;  $W_u = [ ]$ 
4: for  $j = 1 : k$ 
5:   if ( $j == 1$ )
6:      $R_o = Q$ ;  $R_u = 0$ ;  $S_o = \check{Q}$ ;  $S_u = 0$  ▷ Startblöcke
7:   else if ( $j == 2$ )
8:      $R_u = R_o$ ;  $R_o = P_1 R_o$  ▷ Berechnung weiterer Blöcke
9:      $S_u = S_o$ ;  $S_o = \check{P}_1 S_o$ 
10:  else
11:     $R_{old} = R_o$ ;  $S_{old} = S_o$  ▷ Berechnung weiterer Blöcke
12:     $R_o = P_1 R_o + P_2 R_u$ ;  $R_u = R_{old}$ ;  $S_o = \check{P}_1 S_o + \check{P}_2 S_u$ ;  $S_u = S_{old}$ 
13:  end if
14:  for  $i = 1 : \text{Spaltenanzahl}(R_o)$  ▷ Biorthonormalisierung
15:    for  $t = 1 : \text{Spaltenanzahl}(V_o)$  ▷ Biorthogonalisierung
16:       $\delta = [V_o]_{:,t}^* [S_o]_{:,i}$ ;  $\epsilon = [W_o]_{:,t}^* [R_o]_{:,i}$  ▷ Skalarprodukte
17:       $[R_o]_{:,i} = [R_o]_{:,i} - \delta [V_o]_{:,t}$ ;  $[R_u]_{:,i} = [R_u]_{:,i} - \delta [V_u]_{:,t}$ 
18:       $[S_o]_{:,i} = [S_o]_{:,i} - \epsilon [W_o]_{:,t}$ ;  $[S_u]_{:,i} = [S_u]_{:,i} - \epsilon [W_u]_{:,t}$ 
19:    end for
20:     $\delta_o = ||[R_o]_{:,i}||$ ;  $\delta_u = ||[R_u]_{:,i}||$ 
21:     $\epsilon_o = ||[S_o]_{:,i}||$ ;  $\epsilon_u = ||[S_u]_{:,i}||$ 
22:    if ( $\delta_o > \tau$ ) ▷ Deflation des Vektors der Matrix  $V_o$ ?
23:       $v_o = \frac{1}{\delta_o} [R_o]_{:,i}$ ;  $v_u = \frac{1}{\delta_o} [R_u]_{:,i}$  ▷ Normierungen
24:       $V_o = [V_o \ v_o]$ ;  $V_u = [V_u \ v_u]$  ▷ Hinzufügen der Vektoren
25:    else if ( $\delta_u > \tau$ )
26:       $v_o = 0$ ;  $v_u = [R_u]_{:,i}$ 
27:       $V_o = [V_o \ v_o]$ ;  $V_u = [V_u \ v_u]$  ▷ Hinzufügen der Vektoren
28:    else
29:      Deflation
30:    end if
31:    if ( $\epsilon_o > \tau$ ) ▷ Deflation des Vektors der Matrix  $W_o$ ?
32:       $w_o = \frac{1}{\epsilon_o} [S_o]_{:,i}$ ;  $w_u = \frac{1}{\epsilon_o} [S_u]_{:,i}$  ▷ Normierungen
33:       $W_o = [W_o \ w_o]$ ;  $W_u = [W_u \ w_u]$  ▷ Hinzufügen der Vektoren
34:    else if ( $\epsilon_u > \tau$ )
35:       $w_o = 0$ ;  $w_u = [S_u]_{:,i}$ 
36:       $W_o = [W_o \ w_o]$ ;  $W_u = [W_u \ w_u]$  ▷ Hinzufügen der Vektoren
37:    else
38:      Deflation
39:    end if
40:  end for
41: Löschen von Nullspalten der Matrizen  $V_o$  und  $W_o$ 
42: if  $\text{Spaltenanzahl}(V_o) \neq \text{Spaltenanzahl}(W_o)$ 
43:   Ergänzung von  $V_o$  oder  $W_o$  ▷ Um eine identische Anzahl an Basisvektoren zu erhalten
44: end if
45: end for
46:  $V = V_o$ ;  $W = (W_o^* V_o)^{-1} W_o^*$  ▷ Damit gilt:  $W^* V = I$ 

```

Ein Verfahren zur Reduktion von Systemen zweiter Ordnung, das auf der Berechnung einer Basis \mathbf{V} eines Krylov-Unterraumes zweiter Ordnung basiert, wird in [122] vorgeschlagen. Hierbei wird ein Algorithmus angegeben, der auf dem Block-Arnoldi-Algorithmus (Algorithmus 3) für Systeme erster Ordnung basiert, um eine Basis $\tilde{\mathbf{V}}$ des Systems erster Ordnung zu generieren. Anschließend wird eine Basis des Krylov-Unterraumes zweiter Ordnung mit Hilfe von (8.74) definiert. Dabei erfolgt die Orthogonalisierung der Spalten der Spaltenvektoren der Matrix \mathbf{V} in einer nachgelagerten Berechnung mit Hilfe einer \mathbf{QR} -Zerlegung. Der Algorithmus dieses Verfahrens mit einer Entwicklungsstelle s ist in Pseudocode als Algorithmus 8 angegeben. In [15, 155] wer-

Algorithmus 8 Arnoldi-Verfahren für Krylov-Unterräume zweiter Ordnung

Eingabe: Matrizen $\mathbf{M}, \mathbf{D}, \mathbf{K}, \mathbf{F}$; Entwicklungsstelle s ; Stufe des Block-Krylov-Unterraumes k ;

Deflationstoleranz τ

Ausgabe: Basis \mathbf{V}

- 1: function [\mathbf{V}] = Arnoldi_2o($\mathbf{M}, \mathbf{D}, \mathbf{K}, \mathbf{F}, s, k, \tau$)
 - 2: $n = \text{size}(\mathbf{M})$
 - 3: Erzeuge die Matrizen $\mathbf{E}, \mathbf{A}, \mathbf{B}$ eines Systems erster Ordnung der Form (3.5) oder (3.6)
 - 4: Berechne die \mathbf{LU} -Zerlegung: $\mathbf{LU} = (\mathbf{A} - s\mathbf{E})$
 - 5: $\mathbf{P} = \mathbf{U} \setminus (\mathbf{L} \setminus \mathbf{E})$
 - 6: $\mathbf{Q} = -\mathbf{U} \setminus (\mathbf{L} \setminus \mathbf{B})$
 - 7: $\tilde{\mathbf{V}} = \text{Arnoldi}(\mathbf{P}, \mathbf{Q}, k, \tau)$; Algorithmus 3
 - 8: Partitioniere $\mathbf{V}_o = \tilde{\mathbf{V}}(1 : n, :)$
 - 9: Berechne die \mathbf{QR} -Zerlegung: $\mathbf{V}_o = \mathbf{V} \mathbf{R}$
-

den Verfahren vorgeschlagen, die die Transformation in ein System erster Ordnung umgehen und mit Hilfe des Second Order Arnoldi- bzw. den Second Order Lanczos-Algorithmus orthogonale Basen von Krylov-Unterräumen zweiter Ordnung der Form (8.71) direkt erzeugen. Algorithmus 9 stellt den Ablauf des Second Order Arnoldi-Verfahrens anhand eines zweiseitigen Verfahrens zur Reduktion eines Systems zweiter Ordnung der Form (3.1) mit $\mathbf{C}_v = \mathbf{0}$ in Pseudocode dar. Einen Überblick und ein Vergleich verschiedener Verfahren zur Modellreduktion von Systemen zweiter Ordnung kann [109] entnommen werden.

8.4.4.5 Erzielung reellwertiger Systeme

Wie bei Verfahren zur Reduktion von Systemen erster Ordnung ist eine gewünschte Reellwertigkeit der erzeugten Projektionsmatrizen bei Verfahren für Systeme zweiter Ordnung, die mit komplexen Entwicklungsstellen ($s_i \in \mathbb{C}$ mit $\text{Im}(s_i) \neq 0$) verwendet werden, im allgemeinen nicht gegeben. Diese Eigenschaft erhält man, wie bei den Verfahren für Systeme erster Ordnung bereits dargestellt wurde, erst durch Anwendung geeigneter Methoden.

Die Methode für Systeme erster Ordnung, die darauf beruht, dass neben den Entwicklungsstellen $s_i, i = 1, \dots, \hat{i}$ auch implizit die konjugiert komplexen Entwicklungsstellen $\bar{s}_i, i = 1, \dots, \hat{i}$ verwendet werden, kann auf Systeme zweiter Ordnung übertragen werden. Man betrachtet dazu den mit $\tilde{\mathbf{K}}$ und $\tilde{\mathbf{D}}$ wie in (8.80) erhaltenen Block-Krylov-Unterraum für die Entwicklungsstelle

Algorithmus 9 Zweiseitiges Rational Second Order Arnoldi-Verfahren

Eingabe: Matrizen M, D, K, F, C_p ; Vektor der Entwicklungsstellen s ; Vektoren mit Stufen der Block-Krylov-Unterräume k_b und k_c ; Deflationstoleranz τ

Ausgabe: Basen V und W

```

1: function [V, W] = Rational_Arnoldi_2o(M, D, K, F, C_p, s, k_b, k_c, tau)
2:   i = length(s)
3:   V = [], W = []
4:   for i = 1 : i
5:     Berechne die LU-Zerlegung: LU = ([s]_i^2 M + [s]_i D + K)
6:     D_tilde = (2 [s]_i M + D)
7:     P_1 = -U \ (L \ D_tilde)
8:     P_2 = -U \ (L \ M)
9:     Q = U \ (L \ F)
10:    [V_i] = SOAR(P_1, P_2, Q, [k_b]_i, tau); Algorithmus 6 so modifiziert, dass V_i orthogonal zu V ist
11:    P_1 = -L* \ (U* \ D_tilde*)
12:    P_2 = -L* \ (U* \ M*)
13:    Q = L* \ (U* \ C_p*)
14:    [W_i] = SOAR(P_1, P_2, Q, [k_c]_i, tau); Algorithmus 6 so modifiziert, dass W_i orthogonal zu W ist
15:    V = [V V_i], W = [W W_i]
16:    if Spaltenanzahl(V) ≠ Spaltenanzahl(W)
17:      Ergänzung von V oder W, um identische Anzahl an Basisvektoren zu erhalten
18:    end if
19:  end for

```

$s_i \in \mathbb{C}$. Mit (8.60) erhält man diesen mit

$$\tilde{\mathcal{K}}_k \left(\underbrace{\begin{bmatrix} -\tilde{K}_i^{-1} \tilde{D}_i & -\tilde{K}_i^{-1} M \\ I & 0 \end{bmatrix}}_{\Theta(s_i)}, \begin{bmatrix} -\tilde{K}_i^{-1} F \\ 0 \end{bmatrix} \right) = \text{colspan}(V). \quad (8.85)$$

Liegt während der Berechnung der Projektionsmatrix V diese reellwertig vor, dann sind wegen $\bar{\Theta}(s_i) = \Theta(\bar{s}_i)$ in (8.85) die Beziehungen

$$\begin{aligned} \text{Re}(\Theta(s_i) \cdot V) &= \frac{1}{2} ((\Theta(s_i) + \Theta(\bar{s}_i)) \cdot V), \\ \text{Im}(\Theta(s_i) \cdot V) &= \frac{1}{2i} ((\Theta(s_i) - \Theta(\bar{s}_i)) \cdot V) \end{aligned} \quad (8.86)$$

gültig. Es sind also zwei komplexwertige Basisvektoren durch zwei reellwertige Basisvektoren ersetzbar, ohne den aufgespannten Krylov-Unterraum zu verändern. In Analogie erhält man die gleiche Aussage für den Ausgangs-Krylov-Unterraum (8.67), ähnliche Methoden sind auch in [81, 140] zu finden.

Eine weitere Möglichkeit, eine reellwertige Projektionsmatrix zu erhalten, besteht in der Aufteilung der komplexen Projektionsmatrix $V \in \mathbb{C}^{n \times r}$ in ihren Real- und Imaginärteil

$$V_{real} = [\text{Re}([V]_{*1}) \cdots \text{Re}([V]_{*r})] \quad \text{und} \quad V_{imag} = [\text{Im}([V]_{*1}) \cdots \text{Im}([V]_{*r})] \quad (8.87)$$

und anschließender Reorthogonalisierung der Spaltenvektoren der Matrizen V_{real} und V_{imag}

z.B. mit einer reduzierten QR -Zerlegung

$$\mathbf{V} = \text{qr}([\mathbf{V}_{real} \ \mathbf{V}_{imag}]). \quad (8.88)$$

die nach der Berechnung von $2r$ orthogonalen Spalten der Matrix \mathbf{Q} abbricht. Bei einem zweiseitigen Verfahren erhält man eine reellwertige Projektionsmatrix \mathbf{W} durch analoges Vorgehen. Bei Anwendung dieser Methode ist zu beachten, dass sich die Dimension des reduzierten Systems verdoppelt, ohne dass sich die Anzahl der Momente, die mit den Momenten der Übertragungsfunktion des Originalsystems übereinstimmen, ändert.

8.4.4.6 Ungedämpfte und proportional gedämpfte Systeme

In diesem Abschnitt werden die Spezialfälle ungedämpfter und proportional gedämpfter Systeme diskutiert. Wie in [54, 20] wird die Diskussion anhand von Eingangs-Krylov-Unterräumen durchgeführt, sie lässt sich in gleicher Weise jedoch auch auf den Ausgangs-Krylov-Unterraum übertragen. Die Darstellungen dieses Abschnittes orientieren sich dabei an [117].

In [54] wird gezeigt, dass im Fall eines ungedämpften Systems und unter der Voraussetzung, dass die Stufe k des Krylov-Unterraumes zweiter Ordnung ohne Rest halbierbar ist, man für die Entwicklungsstelle $s_i = 0$ den Eingangs-Krylov-Unterraum

$$\begin{aligned} \mathcal{G}_k(\mathbf{0}, -\mathbf{K}^{-1}\mathbf{M}, \mathbf{K}^{-1}\mathbf{F}) &= \text{colspan}\{\mathbf{K}^{-1}\mathbf{F}, \mathbf{0}, -\mathbf{K}^{-1}\mathbf{M}\mathbf{K}^{-1}\mathbf{F}, \dots, \mathbf{0}, (-\mathbf{K}^{-1}\mathbf{M})^{k-1}\mathbf{K}^{-1}\mathbf{F}\} \\ &= \mathcal{K}_{k/2}(-\mathbf{K}^{-1}\mathbf{M}, \mathbf{K}^{-1}\mathbf{F}) \end{aligned} \quad (8.89)$$

erhält.

Betrachtet man beliebige Entwicklungsstellen $s_i \in \mathbb{C}$, so folgt für ein ungedämpftes System kein direkter Zusammenhang zwischen dem Krylov-Unterraum zweiter Ordnung und den klassischen Krylov-Unterräumen wie in (8.89). In [54, 118] wird jedoch gezeigt, dass in diesem Fall mit der Gleichung

$$\begin{aligned} \mathcal{G}_k(-(s_i^2\mathbf{M} + \mathbf{K})^{-1}(2s_i\mathbf{M}), -(s_i^2\mathbf{M} + \mathbf{K})^{-1}\mathbf{M}, (s_i^2\mathbf{M} + \mathbf{K})^{-1}\mathbf{F}) = \\ \mathcal{K}_k(-(s_i^2\mathbf{M} + \mathbf{K})^{-1}\mathbf{M}, (s_i^2\mathbf{M} + \mathbf{K})^{-1}\mathbf{F}) \end{aligned} \quad (8.90)$$

eine ähnliche Beziehung besteht. Aus der Gleichung (8.90) ist nun ersichtlich, dass bei ungedämpften Systemen anstelle eines Krylov-Unterraumes zweiter Ordnung auch ein Krylov-Unterraum erster Ordnung verwendet werden kann. Werden zudem rein imaginäre Entwicklungsstellen s_i verwendet, so kann darüber hinaus vollständig auf komplexe Arithmetik verzichtet werden, da auf der rechten Seite der Gleichung ausschließlich s_i^2 auftritt.

Wie in Abschnitt 4.3 bereits dargestellt, ist im Falle der Rayleigh-Dämpfung die Dämpfungsmatrix als

$$\mathbf{D} = \alpha\mathbf{M} + \beta\mathbf{K} \quad (8.91)$$

definiert. In [54] wird gezeigt, dass der Eingangs-Krylov-Unterraum zweiter Ordnung für die Entwicklungsstelle $s_i = 0$ unabhängig von den Koeffizienten α und β ist. Daher kann dieser durch den Krylov-Unterraum des zugehörigen ungedämpften Systems ausgedrückt werden. Man

erhält analog zu (8.89) für $\alpha = 0$ und gerader Stufe k des Krylov-Unterraumes zweiter Ordnung, die Beziehung

$$\mathcal{G}_k(-\tilde{\mathbf{K}}^{-1}(\beta\mathbf{K}), -\tilde{\mathbf{K}}^{-1}\mathbf{M}, \tilde{\mathbf{K}}^{-1}\mathbf{F}) = \mathcal{K}_{k/2}(-\tilde{\mathbf{K}}^{-1}\mathbf{M}, \tilde{\mathbf{K}}^{-1}\mathbf{F}) \quad (8.92)$$

und für $\alpha \neq 0$ die Beziehung

$$\mathcal{G}_k(-\tilde{\mathbf{K}}^{-1}(\alpha\mathbf{M} + \beta\mathbf{K}), -\tilde{\mathbf{K}}^{-1}\mathbf{M}, \tilde{\mathbf{K}}^{-1}\mathbf{F}) = \mathcal{K}_k(-\tilde{\mathbf{K}}^{-1}\mathbf{M}, \tilde{\mathbf{K}}^{-1}\mathbf{F}). \quad (8.93)$$

Betrachtet man den Momentenabgleich um eine Entwicklungsstelle $s_i \neq 0$, so ist der Eingangs-Krylov-Unterraum nicht mehr invariant gegenüber der Parametrisierung der Dämpfungsmatrix (8.91). Dennoch ist es immer noch möglich, anstelle eines Krylov-Unterraumes zweiter Ordnung einen klassischen Krylov-Unterraum zu verwenden. Die Matrizen $\tilde{\mathbf{D}}$ und $\tilde{\mathbf{K}}$ aus (8.80) der geschifteten Übertragungsfunktion (8.79) ergeben sich für den Fall der Rayleigh-Dämpfung zu

$$\tilde{\mathbf{D}} = (2s_i + \alpha)\mathbf{M} + \beta\mathbf{K} \quad \text{und} \quad \tilde{\mathbf{K}} = s_i^2 + s_i\alpha\mathbf{M} + (s_i\beta + 1)\mathbf{K}. \quad (8.94)$$

Durch eine Umformung der Matrix $\tilde{\mathbf{D}}$ erhält man

$$\tilde{\mathbf{D}} = \tilde{\alpha}\mathbf{M} + \tilde{\beta}\mathbf{K}, \quad (8.95)$$

wobei $\tilde{\alpha} = 2s_i + \alpha - (\beta(s_i^2 + s_i\alpha))/(s_i\beta + 1)$ und $\tilde{\beta} = \beta/(s_i\beta + 1)$, siehe [118]. Damit kann das System zweiter Ordnung wieder als System mit Rayleigh-Dämpfung interpretiert werden, so dass mit (8.93) gilt

$$\mathcal{G}_k(-\tilde{\mathbf{K}}^{-1}\tilde{\mathbf{D}}, -\tilde{\mathbf{K}}^{-1}\mathbf{M}, \tilde{\mathbf{K}}^{-1}\mathbf{F}) = \mathcal{K}_k(-\tilde{\mathbf{K}}^{-1}\mathbf{M}, \tilde{\mathbf{K}}^{-1}\mathbf{F}), \quad (8.96)$$

siehe auch [54, 20].

Mit den Darstellungen dieses Abschnittes kann zusammenfassend festgestellt werden, dass für Systeme ohne Dämpfung oder mit Rayleigh Dämpfung auf das Konzept der Krylov-Unterräume zweiter Ordnung verzichtet werden kann, da sich die entsprechenden Unterräume auch mit Hilfe klassischer Krylov-Unterräume beschreiben lassen. Das bedeutet, dass zur Berechnung der benötigten Projektionsmatrizen auf Methoden, die auf klassischen Krylov-Unterräumen basieren, zurück gegriffen werden kann. Dies ist für die Verfahren, die in dieser Arbeit entwickelt werden, von Bedeutung, da die Methoden, die auf klassischen Krylov-Unterräumen basieren, Vorteile gegenüber den Methoden, die auf Krylov-Unterräumen zweiter Ordnung basieren, aufweisen. Diese Vorteile liegen in der weniger aufwendigen Berechnung der Projektionsmatrizen. Zudem sind die Verfahren numerisch robuster und besser untersucht.

8.4.5 Diskussion der Krylov-Unterraum-Verfahren

Verfahren basierend auf Momentenabgleich und der Verwendung von Krylov-Unterräumen haben sich zur Reduktion von sehr großen Modellen als geeignet erwiesen. Allerdings bestehen bei der Anwendung dieser Verfahren noch Probleme, die bislang noch nicht theoretisch fundiert beantwortet werden konnten.

8.4.5.1 Passivitäts- und Stabilitätserhaltung

Wichtige Eigenschaften eines Systems sind die Passivität und die Stabilität. Deshalb sollte, bevor die Approximationseigenschaften eines Reduktionsverfahrens diskutiert wird, dessen Eigenschaften bezüglich der Stabilitäts- und der Passivitätserhaltung diskutiert werden.

Die Eigenschaft der Erhaltung der Stabilität kann bei Krylov-Unterraum-Verfahren im allgemeinen nicht garantiert werden. Eine einfache und intuitive Herangehensweise dieses Problem zu lösen, besteht in der schrittweisen Erhöhung der Dimension des reduzierten Systems, bis ein stabiles reduziertes System erzeugt wird.

Nach Kenntnis des Autors wurde in der Literatur bis heute keine Methode veröffentlicht, die das Problem der Passivitäts- und Stabilitätserhaltung allgemein gültig und nur mit der Voraussetzung, das ein passives und stabiles Originalsystem vorliegt, löst. Dennoch werden in der Literatur verschiedene Ansätze verfolgt, um sich einer Lösung dieses Problems mit verschiedenen Ansätzen zu nähern. Die drei wichtigsten Ansätze sollen hier kurz beschrieben werden.

Als erstes sind die passivitätserhaltenden Methoden [68, 123, 137] zu nennen. Bei den Verfahren basierend auf diesen Methoden handelt es sich um einseitige Verfahren, die Passivitätserhaltung und damit einhergehend auch Stabilitätserhaltung des reduzierten Systems garantieren, indem die Projektionsmatrix eine Basis eines speziellen Block-Krylov-Unterraumes darstellt. Um eine effiziente Berechnung des reduzierten Systems zu gewährleisten, nutzen diese Verfahren oft die speziellen Strukturen der Systemmatrizen des Originalsystems aus, wodurch deren Anwendungsgebiet eingeschränkt wird.

Einen weiterer Ansatz ist in Methoden zu sehen, die instabile Pole eines reduzierten Systems eliminieren. Bei diesen Methoden werden entweder mit einem nachgelagerten Prozess die instabilen Pole des reduzierten Systems eliminiert oder es werden sogenannte *restarted* Algorithmen, wie der implizit oder explizit *restarted Arnoldi*- oder *restarted Lanczos-Algorithmus* verwendet [12, 82, 101], um ein instabil reduziertes System in ein stabiles System zu transformieren.

Die dritte Gruppe sind die auf Interpolation basierenden Methoden, die die Passivität des Originalsystems durch die Interpolation der sogenannten *spectral zeros* erhalten [4, 5, 9, 96, 167].

Als Nachteile der passivitätserhaltenden Methoden können deren Anwendbarkeit auf eine nur beschränkte Menge von Systemen und ihre Anwendungsmöglichkeit auf ausschließlich einseitige Verfahren gesehen werden.

Der Nachteil der Methoden, die auf *restarted* Algorithmen basieren liegt darin, dass der Aufwand der Implementierung des Algorithmus relativ hoch ist, so dass diese zumeist nur für Verfahren zur Reduktion von SISO-Systemen implementiert werden. Zudem kann bei dieser Methode nach der Entfernung von instabilen Polen die Eigenschaft des Momentenabgleiches und das Auffinden eines stabilen reduzierten Systems nach einer endlichen Anzahl von *restarts* nicht garantiert werden. Die Methoden basierend auf der Interpolation weisen keine dieser Nachteile auf, benötigen allerdings einen höheren numerischen Aufwand als die Krylov-Unterraum-Verfahren.

8.4.5.2 Die Wahl geeigneter Entwicklungsstellen

Eine gute Approximation des Originalsystems ist im hohen Maße abhängig von der Wahl der verwendeten Entwicklungsstellen. Dabei ist die Wahl der Anzahl als auch die Lage der Entwicklungsstellen von Bedeutung.

Eine ausführliche Beschreibung der Auswirkungen der Wahl der Entwicklungsstellen ist in [81, 80] nachzulesen. Hinsichtlich der Wahl der Entwicklungsstellen können die folgenden Grundaussagen gemacht werden, die für alle Rational-Krylov-Verfahren zutreffen:

- Die Verwendung einer Entwicklungsstelle s_0 mit einem kleinem Betrag des Imaginärteils resultiert in einer guten Approximation des unteren Frequenzbandes durch das reduzierte System.
- Die Verwendung einer Entwicklungsstelle s_0 mit einem großen Betrag des Imaginäranteils resultiert in einer guten Approximation des oberen Frequenzbandes durch das reduzierte System.
- Mit der Verwendung ausschließlich rein imaginärer Entwicklungsstellen können sehr gute Approximationen des Frequenzbereiches in der Umgebung der Entwicklungsstellen erreicht werden. Allerdings kann die Konvergenz der reduzierten Modelle gegen das exakte Übertragungsverhalten des Originalsystems für weit von den Entwicklungsstellen entfernte Frequenzbereiche sehr langsam sein.
- Die Verwendung ausschließlich reeller Entwicklungsstellen ermöglicht eine gute Approximation eines größeren Frequenzbereiches als bei Verwendung rein imaginärer Entwicklungsstellen. Die Approximation in der Nähe der Entwicklungsstelle ist allerdings weniger genau als bei der Verwendung einer rein imaginären oder komplexen Entwicklungsstelle.
- Die Verwendung mehrerer Entwicklungsstellen $s_i, i = 1, \dots, \hat{i}$ resultiert in einer guten Approximation eines breiteren Frequenzbandes durch das reduzierte System.

Diese Informationen über die Auswirkungen der Wahl der Entwicklungsstellen kann nur als Ansatz zur Bestimmung guter Entwicklungsstellen dienen, exakte Werte für die optimale Wahl der Entwicklungsstellen sind daraus nicht abzuleiten.

In [81] wird die Wahl der Entwicklungsstellen als Approximation der Eigenwerte des Originalsystem aufgefaßt. Aus den Untersuchungen schlägt der Autor eine Kombination aus rein reellen und imaginären Entwicklungsstellen vor, die linear oder logarithmisch auf der reellen und imaginären Achse der komplexen Ebene verteilt liegen. Mit diesem Ansatz sind mit ausreichend vielen Entwicklungsstellen gute Approximationsergebnisse erzielbar. Allerdings können a priori keine Aussagen über die Anzahl der benötigten Entwicklungsstellen getroffen werden.

In [48] wird eine Methode vorgeschlagen, die den Arnoldi-Algorithmus verwendet, um geeignete Entwicklungsstellen zu finden. Der Arnoldi-Algorithmus wird verwendet, um zu jeder Entwicklungsstelle einer gegebenen Menge von Entwicklungsstellen $s_i, i = 1, \dots, \hat{i}$ jeweils ein reduziertes System der Dimension r_i zu bestimmen. Anschließend wird aus der Menge, die aus den Polen der

Übertragungsfunktionen der reduzierten Modelle besteht, eine Teilmenge verwendet, deren Elemente zur Bestimmung der Entwicklungsstellen für das Rational-Krylov-Unterraum-Verfahren dienen.

8.4.5.3 Die Wahl der geeigneten Stufen der verwendeten Krylov-Unterräume

Neben der optimalen Wahl der Entwicklungsstellen ist auch die Wahl der geeigneten Stufen der verwendeten Krylov-Unterräume zu den gewählten Entwicklungsstellen für die Erzielung eines optimalen Approximationsergebnisses ausschlaggebend.

In Abschnitt 9.3 werden Verfahren zur Automatisierung des Reduktionsprozesses, die auf einer adaptiven Wahl der Entwicklungsstellen aus einer gegebenen Menge von potenziellen Entwicklungsstellen und der Stufen der dazu korrespondierenden Krylov-Unterräume basieren, diskutiert. Nach Kenntnis des Autors existiert bislang jedoch kein Verfahren, das eine optimale Wahl der Entwicklungsstellen mit einer adaptiven Wahl der Stufen der verwendeten Krylov-Unterräume kombiniert.

8.4.5.4 Globale Fehlerschranke

Im Gegensatz zu Verfahren, die auf Gramschen Matrizen basieren, existiert für Reduktionsverfahren, die auf Moment Matching basieren, keine allgemeine, numerisch effizient berechenbare und theoretisch fundierte Schranke für den Approximationsfehler des reduzierten Systems. In [13] wird für das Padé-via-Lanczos-Verfahren eine Fehlerschranke hergeleitet. Allerdings müssen die Systemmatrizen des Originalsystems spezielle Voraussetzungen erfüllen, damit diese Fehlerschranke in der Praxis anwendbar ist. In der Praxis bieten sich daher heuristische Methoden zur Bestimmung des Approximationsfehlers an. In [21] werden zwei unterschiedliche heuristische Methoden vorgestellt, für die allerdings keine theoretischen Beweise angegeben werden können. Die erste Methode nutzt die Erkenntnis, dass der Fehler zwischen dem originalen System und dem reduzierten System annähernd dem Approximationsfehler zwischen zwei reduzierten Systemen der Dimension r und $r + 1$ entspricht. Die zweite Methode basiert auf der Beobachtung, dass sich die Hankel-Singulärwerte des reduzierten Systems mit Erhöhung der Dimension des reduzierten Systems denen des Originalsystems annähern.

8.4.5.5 Adaptive Methoden

Ein weiteres offenes Problem, das mit dem Fehlen einer geeigneten Fehlerschranke zusammen hängt, stellt die Wahl der optimalen Dimension des reduzierten Systems dar.

Eine Methode, die in [154] vorgestellt wird, nutzt die lineare Abhängigkeit der durch den Arnoldi- oder den Lanczos-Algorithmus generierten Vektoren, um ein Abbruchkriterium für die Erzeugung der Projektionsmatrizen herzuleiten.

Weitere Methoden nutzen den Approximationsfehler des reduzierten Systems, um die Ordnung des reduzierten Systems zu erhöhen oder den Reduktionsprozess abubrechen. In Ermangelung einer theoretisch fundierten globalen Fehlerschranke, die eine solche adaptive Anpassung der

Dimension des reduzierten Systems ermöglichen würde, können in der Praxis heuristische Fehlerschätzmethoden herangezogen werden.

8.4.5.6 Systeme mit vielen Ein- und Ausgängen

Bei der Reduktion von Systemen mit einer geringen Anzahl an Ein- und Ausgängen sind durch die Verwendung von Krylov-Unterraum-Verfahren gute Approximationen mit einem verhältnismäßig geringem numerischem Aufwand erzielbar. Bei der Reduktion von Systemen mit sehr vielen Ein- und Ausgängen stellt sich die Situation anders dar. Hier erweisen sich die Verfahren als kaum anwendbar.

Das Problem liegt dabei in der Art der Berechnung der Projektionsmatrizen \mathbf{V} und \mathbf{W} und damit einhergehend in der Ordnung des reduzierten Systems. In jeder Iteration eines Krylov-Unterraum-Verfahrens werden $\max(\sum_{i=1}^{\hat{i}} q \cdot k(i), \sum_{i=1}^{\hat{i}} m \cdot k(i))$ neue Spaltenvektoren der Projektionsmatrix generiert, wenn q die Anzahl der Eingänge, m die Anzahl der Ausgänge, \hat{i} die Anzahl der Entwicklungsstellen und $k(i)$ die Stufe des Krylov-Unterraumes zu jeder Entwicklungsstelle darstellen. Dies führt bei einer großen Anzahl an Ein- bzw. Ausgängen zu einem schnellen Anwachsen der Ordnung des reduzierten Systems. Erste Verfahren, die für eine effiziente Reduktion solcher Systeme entwickelt wurden, werden in [59, 62] vorgestellt.

8.5 \mathcal{H}_2 -optimale Verfahren

Das Problem, reduzierte Systeme mit minimalem Approximationsfehler bezüglich der \mathcal{H}_2 -Norm zu finden, ist eines der am weitesten zurück liegenden Probleme in der Modellreduktion, siehe [17, 69, 89, 157, 168]. Dabei stellt sich das Problem ein globales Optimum zu finden wie folgt dar:

Problem 8.1 *Gegeben sei ein asymptotisch stabiles LTI System Σ der Dimension n mit Übertragungsfunktion \mathbf{H} . Gesucht wird ein stabiles reduziertes System $\hat{\Sigma}$ der Dimension $r \ll n$, welches die \mathcal{H}_2 -Norm des Fehlersystems $\Sigma - \hat{\Sigma}$ minimiert. Es ist somit die Lösung des Minimierungsproblems*

$$\min \mathcal{J}(\hat{\mathbf{H}}) = \|\mathbf{H} - \hat{\mathbf{H}}\|_{\mathcal{H}_2}^2, \quad (8.97)$$

zu finden, d. h. es wird das stabil reduzierte System $\hat{\mathbf{H}}$ der Dimension r gesucht, dessen Approximationsfehler $\mathbf{H} - \hat{\mathbf{H}}$ bezüglich der \mathcal{H}_2 -Norm minimal ist.

Bei den im folgenden dargestellten \mathcal{H}_2 -optimalen Verfahren handelt es sich um spezielle Krylov-Unterraum-Verfahren, bei denen die Wahl der Entwicklungsstellen durch eine spezielle iterative Methode erfolgt. Bis auf die Wahl der Entwicklungsstellen und der Wahl der Dimensionen der dazu korrespondierenden Krylov-Unterräume treffen somit die generellen, im vorhergehenden Abschnitt diskutierten, Eigenschaften der Krylov-Unterraum-Verfahren auch auf diese Verfahren zu.

8.5.1 IRKA

Ein effizientes Verfahren zur Reduktion von SISO-Systemen erster Ordnung ist der sogenannte *iterative rationale Krylov-Algorithmus*⁵ (IRKA), der hier als Algorithmus 10 in Pseudocode dargestellt ist. Mit diesem Algorithmus erhält man bei Konvergenz ein \mathcal{H}_2 -optimales reduziertes System [87].

Im Falle eines SISO-Systems erster Ordnung erzeugt dieses Verfahren bei Konvergenz ein reduziertes Verfahren, dass die Interpolations-basierten \mathcal{H}_2 -Optimalitätsbedingungen erster Art erfüllt. Diese Bedingungen wurden von Meier und Luenberger bereits 1967 hergeleitet [124].

Satz 8.15 (Meier-Luenberger-Bedingungen für SISO-Systeme) *Sei ein asymptotisch stabiles SISO LTI-System mit Übertragungsfunktion $H(s)$ gegeben. Sei $\hat{H}(s)$ eine lokale Lösung des Minimierungsproblems (8.97) unter der Annahme, dass $\hat{H}(s)$ einfache Polstellen λ_k , $k = 1, \dots, r$ hat. Dann interpoliert $\hat{H}(s)$ sowohl $H(s)$ als auch dessen erste Ableitung an der Stelle $-\bar{\lambda}_k$, $k = 1, \dots, r$, d. h. es gilt:*

$$\begin{aligned} H(-\bar{\lambda}_k) &= \hat{H}(-\bar{\lambda}_k), \\ H'(-\bar{\lambda}_k) &= \hat{H}'(-\bar{\lambda}_k), \end{aligned} \tag{8.98}$$

für $k = 1, \dots, r$.

BEWEIS: Siehe [124]. □

Algorithmus 10 Iterativer rationaler Krylov-Algorithmus (IRKA) nach [87]

Eingabe: Matrizen $\mathbf{E}, \mathbf{A}, \mathbf{b}, \mathbf{c}$; Vektor mit Anfangsentwicklungsstellen \mathbf{s}

Ausgabe: Basen \mathbf{V} und \mathbf{W}

- 1: function $[\mathbf{V}, \mathbf{W}] = \text{IRKA}(\mathbf{E}, \mathbf{A}, \mathbf{b}, \mathbf{c}, \mathbf{s})$
 - 2: $r = \text{length}(\mathbf{s})$
 - 3: Wähle \mathbf{V} und \mathbf{W} so, dass $\mathbf{W}^* \mathbf{V} = \mathbf{I}$ und

$$\begin{aligned} \text{colspan}(\mathbf{V}) &= \text{span}\{([s]_1 \mathbf{E} - \mathbf{A})^{-1} \mathbf{b}, \dots, ([s]_r \mathbf{E} - \mathbf{A})^{-1} \mathbf{b}\} \\ \text{colspan}(\mathbf{W}) &= \text{span}\{([s]_1 \mathbf{E}^* - \mathbf{A}^*)^{-1} \mathbf{c}^*, \dots, ([s]_r \mathbf{E}^* - \mathbf{A}^*)^{-1} \mathbf{c}^*\} \end{aligned}$$
 - 4: **while** keine Konvergenz
 - 5: $\hat{\mathbf{E}} = \mathbf{W}^* \mathbf{E} \mathbf{V}$ und $\hat{\mathbf{A}} = \mathbf{W}^* \mathbf{A} \mathbf{V}$
 - 6: Lösung des Eigenwertproblems $\lambda \hat{\mathbf{E}} \mathbf{x} = \hat{\mathbf{A}} \mathbf{x}$
 - 7: $[s]_i \leftarrow -\bar{\lambda}_i$ für $i = 1, \dots, r$
 - 8: Biorthonormalisiere \mathbf{V} und \mathbf{W} so, dass $\mathbf{W}^* \mathbf{V} = \mathbf{I}$ und

$$\begin{aligned} \text{colspan}(\mathbf{V}) &= \text{span}\{([s]_1 \mathbf{E} - \mathbf{A})^{-1} \mathbf{b}, \dots, ([s]_r \mathbf{E} - \mathbf{A})^{-1} \mathbf{b}\} \\ \text{colspan}(\mathbf{W}) &= \text{span}\{([s]_1 \mathbf{E}^* - \mathbf{A}^*)^{-1} \mathbf{c}^*, \dots, ([s]_r \mathbf{E}^* - \mathbf{A}^*)^{-1} \mathbf{c}^*\} \end{aligned}$$
 - 9: **end while**
-

Da die \mathcal{H}_2 -Norm für instabile Systeme nicht definiert ist, sind die Meier-Luenberger-Bedingungen und die \mathcal{H}_2 -optimalen Verfahren nur auf stabile SISO-Systeme mit einfachen Polen anwendbar.

⁵In Englisch: *Iterative Rational Krylov Algorithm*

8.5.2 Tangentiale Hermite-Interpolationsverfahren

Betrachtet man MIMO-Systeme führt die Methode des Moment-Matching zu großen und aufwendig zu berechnenden reduzierten Systemen, da sich die Dimension des reduzierten Systems für jedes Moment um die Anzahl der Übertragungspfade des Systems erhöht.

Aus diesem Grund wurden Verfahren entwickelt, die auf einer tangential hermiten Interpolation der Übertragungsfunktion des Systems basieren und tangentiales rationales Hermite-Interpolationsverfahren genannt werden [178]. Anstelle die Momente der reduzierten Übertragungsfunktion $\hat{\mathbf{H}}$ an verschiedenen Entwicklungsstellen s_i der komplexen Ebene mit den Momenten der originalen Übertragungsfunktion $\mathbf{H}(s)$ zur Übereinstimmung zu bringen, wird bei diesen Verfahren ein reduziertes System ermittelt, dessen links und rechts tangentiale Richtungen der Übertragungsfunktion an den Entwicklungsstellen mit denen des Originalsystems übereinstimmen.

Im Gegensatz zu Verfahren, die auf Moment-Matching basieren, wird auf die Eigenschaft des Moment-Matching der Übertragungsfunktion des reduzierten Systems verzichtet. Bei diesen Verfahren reicht aus, dass die Übertragungsfunktion $\hat{\mathbf{H}}(s)$ des reduzierten Systems eine linksseitig tangential Interpolierende in die Richtungen ℓ_i , eine rechtsseitig tangential Interpolierende in die Richtungen \mathbf{r}_i , und eine zweiseitig tangential Interpolierende der Ableitung in die Richtungen ℓ_i und \mathbf{r}_i der Übertragungsfunktion $\mathbf{H}(s)$ des Originalsystems an den Entwicklungsstellen s_i für $i = 1, 2, \dots, r$ ist:

$$\begin{aligned} \mathbf{H}(s_i) \mathbf{r}_i &= \hat{\mathbf{H}}(s_i) \mathbf{r}_i, \\ \ell_i \mathbf{H}(s_i) &= \ell_i \hat{\mathbf{H}}(s_i), \\ \ell_i \mathbf{H}'(s_i) \mathbf{r}_i &= \ell_i \hat{\mathbf{H}}'(s_i) \mathbf{r}_i. \end{aligned} \tag{8.99}$$

Die durch (8.99) gegebenen Bedingungen an das reduzierte System sind zwar schwächer als die Moment-Matching-Eigenschaften, doch werden bei der praktischen Anwendung dieser Methode neben geeigneten Entwicklungsstellen zusätzlich geeignete tangentiale Richtungen an den Entwicklungsstellen benötigt.

Lemma 8.2 *Seien $\mathbf{V}, \mathbf{W} \in \mathbb{K}^{n \times r}$ Matrizen mit vollem Rang r so, dass $\mathbf{W}^* \mathbf{V} = \mathbf{I}$. Sei $s_i \in \mathbb{K}$, $\ell_i \in \mathbb{K}^{1 \times q}$ und $\mathbf{r}_i \in \mathbb{K}^{m \times 1}$ für $i = 1, \dots, r$ eine gegebene Menge an Entwicklungsstellen sowie linken und rechten tangentialen Richtungsvektoren. Angenommen die Entwicklungsstellen s_i sind so gewählt, dass die Matrizen $(\mathbf{A} - s_i \mathbf{I})$ invertierbar sind und für alle $i = 1, \dots, r$ gilt:*

$$\begin{aligned} (s_i \mathbf{I} - \mathbf{A})^{-1} \mathbf{B} \mathbf{r}_i &\in \text{colspan}(\mathbf{V}), \\ (\bar{s}_i \mathbf{I} - \mathbf{A}^*)^{-1} \mathbf{C}^* \ell_i^* &\in \text{colspan}(\mathbf{W}), \end{aligned} \tag{8.100}$$

dann erfüllt das mit Hilfe der Matrizen \mathbf{V} und \mathbf{W} reduzierte System der Form (5.10) mit $\mathbf{E} = \mathbf{I}$ die Bedingungen (8.99).

BEWEIS: Siehe z. B. [178]. □

8.5.3 MIRIAM

Wie im vorhergehenden Abschnitt dargestellt wurde, kann das reduzierte System durch die Eigenschaften (8.99) festgelegt werden. Wie im Fall von Verfahren, die auf Moment-Matching basieren, ist die explizite Berechnung der Momente 0-ter Ordnung an den Entwicklungsstellen numerisch instabil. Um diese Momente mit den geforderten Eigenschaften der tangential hermites Interpolation der Übertragungsfunktion numerisch stabil zu berechnen, werden diese deshalb implizit mit Krylov-Unterraum-Verfahren erzeugt.

Da die optimalen Entwicklungsstellen und dazu tangentialen Richtungsvektoren a priori nicht bekannt sind, werden in [32, 87, 180] Methoden vorgestellt, die auf der tangentialen hermites Interpolation basieren und die Entwicklungsstellen sowie die tangentialen Richtungsvektoren iterativ bestimmen. In [32, 111] wird ein effizientes iteratives Verfahren für MIMO-Systeme, der *MIMO iterative rationale Interpolations-Algorithmus*⁶ (MIRIAM) genannt wird und als Erweiterung von IRKA für MIMO-Systeme angesehen werden kann, vorgestellt. Der Ablauf dieses Verfahrens ist als Algorithmus 11 dargestellt, wobei der hier angegebene Algorithmus einen zur Reduktion von MIMO-Systemen in allgemeiner Zustandsdarstellung angepassten Ablauf des Verfahrens darstellt.

Algorithmus 11 MIMO iterativer rationaler Interpolations-Algorithmus (MIRIAM) nach [32, 111]

Eingabe: Matrizen E, A, B, C ; Vektor mit Anfangsentwicklungsstellen $s = \{s_1, \dots, s_r\}$; Anfangsrichtungsvektoren $\ell = \{\ell_1, \dots, \ell_r\}$ und $r = \{r_1, \dots, r_r\}$

Ausgabe: Basen V und W

```

1: function [  $V, W$  ] = MIRIAM( $E, A, B, C, s, \ell, r$ )
2:   while keine Konvergenz
3:      $V = [], W = []$ 
4:     for  $k = 1 : r$ 
5:        $v_k = (s_k E - A)^{-1} B r_k^*, V = [V \ v_k]$ 
6:        $w_k = (\bar{s}_k E^* - A^*)^{-1} C^* \ell_k, W = [W \ w_k]$ 
7:     end for
8:      $W = (W^* V)^{-1} W^*$ , damit  $W^* V = I$ 
9:      $\hat{E} = W^* E V$ ;  $\hat{A} = W^* A V$ ;  $\hat{B} = W^* B$ ;  $\hat{C} = C V$ 
10:    Lösung des Eigenwertproblems  $\lambda \hat{E} x = \hat{A} x$  mit den Eigenwerten  $\{s_1, \dots, s_r\}$ , der Matrix  $Y$  der linken
        Eigenvektoren  $Y = [y_1 \cdots y_r]$  und der Matrix  $X$  der rechten Eigenvektoren  $X = [x_1 \cdots x_r]$ 
11:    Wähle neue Entwicklungsstellen  $s_i \leftarrow -\bar{\lambda}_i$  für  $i = 1, \dots, r$ , neue linke Richtungsvektoren mit  $\ell_k$  als  $k$ -te
        Spalte von  $(\hat{C} Y)^*$  und neue rechte Richtungsvektoren mit  $r_k$  als  $k$ -te Spalte von  $(X \hat{B})^*$ 
12:   end while
    
```

⁶In Englisch: *MIMO Iterative Rational Interpolation Algorithm*

Kapitel 9

Fehlerbestimmung und adaptive Methoden

Dieses Kapitel soll eine Übersicht und eine kurze Abwägung der Vor- bzw. Nachteile bestehender Methoden zur Fehlerbestimmung Krylov-Unterraum-reduzierter Systeme geben, die für die weiteren Überlegungen in dieser Arbeit relevant sind. Im Fokus der Betrachtungen liegt dabei die Bestimmung des Approximationsfehlers eines reduzierten Systems, der bei adaptiven Methoden als Entscheidungskriterium zur Anpassung der Parameter von Reduktionsverfahren verwendet werden kann und damit eine Automatisierung des Reduktionsprozesses ermöglicht.

Die verschiedenen Methoden zur Fehlerbestimmung eines reduzierten Modells lassen sich in die zwei Hauptgruppen der analytischen und der heuristischen Methoden unterteilen. Diese sind wiederum in lokale und globale Methoden gliedern. Im folgenden Schema sind diese Gliederung, der Methoden und die Abschnitte in denen bestehende Methoden näher beschrieben werden, dargestellt. Die verschiedenen Methoden zur Bestimmung des Approximationsfehlers werden

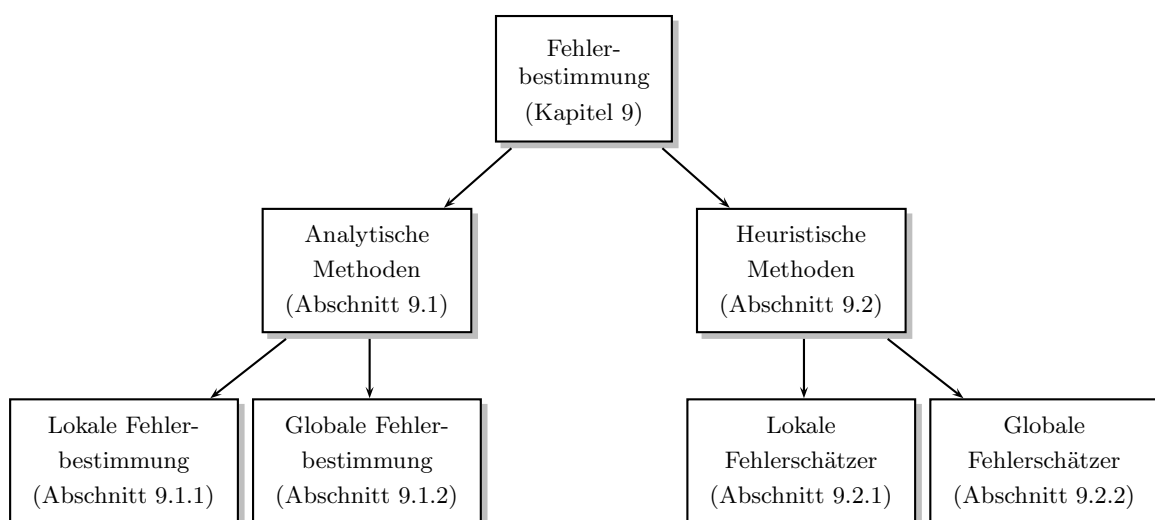


Abbildung 9.1: Schema zur Kategorisierung von Fehlerbestimmungsmethoden

hier ausschließlich anhand von Systemen erster Ordnung dargestellt, da die Bestimmung des Approximationsfehlers für Systeme zweiter Ordnung nach einer Transformation des Systems

zweiter Ordnung in ein System erster Ordnung möglich ist.

Im abschließenden Abschnitt (9.3) dieses Kapitels wird eine bestehende Methode zur adaptiven Auswahl der Entwicklungsstellen aus einer vorgegebenen Menge von potenziellen Entwicklungsstellen und eine Methode zur adaptiven Bestimmung der Stufen der zu den Entwicklungsstellen korrespondierenden Krylov-Unterräumen vorgestellt.

9.1 Analytische Methoden

In diesem Abschnitt werden existierende Methoden zur analytischen Bestimmung von Fehler-schranken dargestellt. Zu Beginn erfolgt eine Beschreibung einer Methode zur lokalen Fehlerbestimmung. Darauf folgt ein Abschnitt, in dem eine globale Fehlerbestimmung zwischen dem reduzierten und dem Originalsystems diskutiert wird.

9.1.1 Lokale Fehlerbestimmung

Bei der lokalen Fehlerbestimmung lassen sich unterschiedliche Fehlermaße betrachten. In dieser Arbeit werden dazu zwei Fehlermaße näher dargestellt. Dabei handelt es sich zum einen um den Approximationsfehler der Übertragungsfunktion und zum anderen um den Approximationsfehler des k -ten Momentes der Übertragungsfunktion an einer Stelle s_0 .

Bei der Verwendung des Approximationsfehler der Übertragungsfunktion eines reduzierten (MIMO-) Systems als Fehlermaß wird der absolute Approximationsfehler $\epsilon_{abs}(s_0) = \|\epsilon(s_0)\| = \|\mathbf{H}(s_0) - \hat{\mathbf{H}}(s_0)\|$ eines reduzierten Systems der Dimension r an der Stelle s_0 bezüglich einer geeigneten Norm $\|\cdot\|$ betrachtet.

Wird der der Approximationsfehler des k -ten Momentes der Übertragungsfunktion als Fehlermaß verwendet, so wird der (absolute) Approximationsfehler $\epsilon_k(s_0) = \|\epsilon_k(s_0)\| = \|\mathbf{H}_k(s_0) - \hat{\mathbf{H}}_k(s_0)\|$ eines reduzierten Systems der Dimension r an der Stelle s_0 bezüglich einer geeigneten Norm $\|\cdot\|$ betrachtet.

9.1.1.1 Fehlerbestimmung der Übertragungsfunktion

In [81] wird eine Berechnung des Approximationsfehlers eines reduzierten SISO-Systems erster Ordnung (3.4) über die Berechnung des Residuums an einer Stelle s im Frequenzgang beschrieben. Dabei wird eine mögliche Bestimmung des skalaren Approximationsfehlers $\epsilon(s_0) = H(s_0) - \hat{H}(s_0)$ eines reduzierten SISO-Systems der Dimension r an der Stelle s_0 durch den folgenden Satz motiviert.

Satz 9.1 (Fehler der Übertragungsfunktion eines reduzierten SISO-Systems) *Der Fehler der Übertragungsfunktion $\epsilon(s_0) = H(s_0) - \hat{H}(s_0)$ eines SISO-Systems erster Ordnung an der Stelle s_0 ist durch*

$$\epsilon(s_0) = \mathbf{r}_c^*(s_0)(\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{r}_b(s_0) \quad (9.1)$$

gegeben, wobei $\mathbf{r}_b(s_0) = (\mathbf{A} - s_0\mathbf{E})\tilde{\mathbf{x}} - \mathbf{b}$ und $\mathbf{r}_c(s_0) = (\mathbf{A} - s_0\mathbf{E})\tilde{\mathbf{z}} - \mathbf{c}^$. Dabei sind $\tilde{\mathbf{x}} = \mathbf{V}(\hat{\mathbf{A}} - s_0\hat{\mathbf{E}})^{-1}\hat{\mathbf{b}}$ und $\tilde{\mathbf{z}} = \mathbf{W}(\hat{\mathbf{A}} - s_0\hat{\mathbf{E}})^{-*}\hat{\mathbf{c}}^*$ die durch Projektion erhaltenen Approximationen*

der Residuen $\mathbf{x} = (\mathbf{A} - s_0 \mathbf{E})^{-1} \mathbf{b}$ bzw. $\mathbf{z} = (\mathbf{A} - s_0 \mathbf{E})^{-T} \mathbf{c}^*$.

BEWEIS: Siehe [80]. □

Um den Fehler an einer Stelle s_0 auszuwerten, muss die rechte Seite der Gleichung

$$\varepsilon(s_0) = \mathbf{r}_c(s_0)^* (\mathbf{A} - s_0 \mathbf{E})^{-1} \mathbf{r}_b(s_0) \quad (9.2)$$

mit

$$\mathbf{r}_b(s_0) = (\mathbf{A} - s_0 \mathbf{E}) \mathbf{V} (\hat{\mathbf{A}} - s_0 \hat{\mathbf{E}})^{-1} \hat{\mathbf{b}} - \mathbf{b} \quad (9.3)$$

und

$$\mathbf{r}_c(s_0) = (\mathbf{A} - s_0 \mathbf{E})^* \mathbf{W} (\hat{\mathbf{A}} - s_0 \hat{\mathbf{E}})^{-*} \hat{\mathbf{c}}^* - \mathbf{c}^* \quad (9.4)$$

berechnet werden.

Anmerkung 9.1 Da in dieser Arbeit MIMO-Systeme betrachtet werden, wird hier die Aussage des Satzes 9.1 auf MIMO-Systeme übertragen. Um den Fehler an einer Stelle s_0 auszuwerten, muss die rechte Seite der Gleichung

$$\epsilon(s_0) = \|\mathbf{R}_C(s_0)^* (\mathbf{A} - s_0 \mathbf{E})^{-1} \mathbf{R}_B(s_0)\| \quad (9.5)$$

mit

$$\mathbf{R}_B(s_0) = (\mathbf{A} - s_0 \mathbf{E}) \mathbf{V} (\hat{\mathbf{A}} - s_0 \hat{\mathbf{E}})^{-1} \hat{\mathbf{B}} - \mathbf{B} \quad (9.6)$$

und

$$\mathbf{R}_C(s_0) = (\mathbf{A} - s_0 \mathbf{E})^* \mathbf{W} (\hat{\mathbf{A}} - s_0 \hat{\mathbf{E}})^{-*} \hat{\mathbf{C}}^* - \mathbf{C}^* \quad (9.7)$$

berechnet werden, wobei $\|\cdot\|$ eine geeignete Norm darstellt. □

Die Ermittlung des Approximationsfehlers eines reduzierten Systems mit dieser Methode ist aufwendig zu berechnen, da zur Berechnung des Fehlers an jeder Stelle s_0 ein Gleichungssystem bzw. ein Gleichungssystem mit mehreren rechten Seiten der Dimension des Systems zu lösen ist. Die Berechnung des Fehlers innerhalb eines großen Frequenzintervalls ist mit dieser Methode deshalb kaum durchführbar, da zur Berechnung eines aussagefähigen Fehlerverlaufes zwangsweise die Fehler an zahlreichen Stellen ausgewertet werden müssen.

9.1.1.2 Fehlerbestimmung des k -ten Momentes der Übertragungsfunktion

Neben dem Approximationsfehler der Übertragungsfunktion kann auch der Approximationsfehler des k -ten Momentes der Übertragungsfunktion als Fehlermaß betrachtet werden, siehe z.B. [80].

Definition 9.1 (k -ter Momentenfehler eines reduzierten SISO-Systems) Der Fehler $\varepsilon_k(s_0)$ des k -ten Momentes der Übertragungsfunktion $\hat{H}(s)$ eines reduzierten SISO-Systems an der Entwicklungsstelle s_0 ist durch

$$\varepsilon_k(s_0) := H_k(s_0) - \hat{H}_k(s_0) \quad (9.8)$$

$$= -\mathbf{c}((\mathbf{A} - s_0 \mathbf{E})^{-1} \mathbf{E})^j (\mathbf{A} - s_0 \mathbf{E})^{-1} \mathbf{b} - \left[-\hat{\mathbf{c}}((\hat{\mathbf{A}} - s_0 \hat{\mathbf{E}})^{-1} \hat{\mathbf{E}})^j (\hat{\mathbf{A}} - s_0 \hat{\mathbf{E}})^{-1} \hat{\mathbf{b}} \right] \quad (9.9)$$

gegeben, wobei $H_k(s_0)$ und $\hat{H}_k(s_0)$ die k -ten Momente der Übertragungsfunktionen des unreduzierten bzw. des reduzierten SISO-Systems bezeichnen.

Da in dieser Arbeit MIMO-Systeme betrachtet werden, wird hier zudem der k -te Momentenfehler der Übertragungsfunktion für reduzierte MIMO-Systeme definiert.

Definition 9.2 (k -ter Momentenfehler eines reduzierten MIMO-Systems) Der Fehler $\varepsilon_k(s_0)$ des k -ten Momentes der Übertragungsfunktion $\hat{\mathbf{H}}(s)$ eines reduzierten MIMO-Systems an der Entwicklungsstelle s_0 ist durch

$$\varepsilon_k(s_0) := \mathbf{H}_k(s_0) - \hat{\mathbf{H}}_k(s_0) \quad (9.10)$$

$$= -\mathbf{C}((\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{E})^j(\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{B} - \left[-\hat{\mathbf{C}}((\hat{\mathbf{A}} - s_0\hat{\mathbf{E}})^{-1}\hat{\mathbf{E}})^j(\hat{\mathbf{A}} - s_0\hat{\mathbf{E}})^{-1}\hat{\mathbf{B}} \right] \quad (9.11)$$

gegeben, wobei $\mathbf{H}_k(s_0)$ und $\hat{\mathbf{H}}_k(s_0)$ die k -ten Momente der Übertragungsfunktionen des unreduzierten bzw. des reduzierten MIMO-Systems bezeichnen.

Auch die Ermittlung dieses Fehlermaßes ist aufwendig zu berechnen, da zu dessen Berechnung an jeder Stelle s_0 ein Gleichungssystem bzw. ein Gleichungssystem mit mehreren rechten Seiten der Dimension des Systems zu lösen ist.

9.1.2 Globale Fehlerbestimmung

In [10] wird eine mögliche globale Fehlerbestimmung eines durch balancierten Abschneidens reduzierten Systems vorgestellt. Dazu werden die Hankel-Singulärwerte σ_i , $i = 1, \dots, 2n$ des Originalsystems betrachtet. Der Fehler, gemessen in der \mathcal{H}_∞ -Norm, ist dann über die Gleichung

$$\|\mathbf{H} - \hat{\mathbf{H}}\|_{\mathcal{H}_\infty} \leq 2(\sigma_{r+1} + \dots + \sigma_{2n}), \quad (9.12)$$

wobei \mathbf{H} die Übertragungsfunktion des originalen und $\hat{\mathbf{H}}$ die Übertragungsfunktionen des reduzierten Systems der Dimension r darstellen, ermittelbar.

Der Vorteil dieser Methode ist die Bestimmung des exakten Approximationsfehlers des reduzierten Systems. Allerdings ist diese Methode zur Fehlerbestimmung nur für reduzierte Systeme, die mit Verfahren des balancierten Abschneidens erzeugt wurden, verwendbar.

\mathcal{H}_2 -Fehlerbestimmung

In [87] wird eine Methode vorgestellt, um den \mathcal{H}_2 -Fehler eines reduzierten SISO-Systems zu bestimmen. Diese Methode ist für reduzierte Modelle gültig, deren Momente mit den ersten $2r$ Momenten des Originalsystems übereinstimmen.

Seien dazu $H(s)$ und $\hat{H}(s)$ die Übertragungsfunktionen des Original- bzw. des reduzierten SISO-Systems. Weiterhin seien Φ_i und $\hat{\Phi}_i$ die Residuen der Übertragungsfunktionen $H(s)$ und $\hat{H}(s)$ an den Polstellen λ_i und $\hat{\lambda}_i$ des Originalsystems bzw. des reduzierten Systems:

$$\Phi_i = \text{res}(H(s), \lambda_i) = H(s - \lambda_i)|_{s=\lambda_i}, \quad i = 1, \dots, 2n \quad \text{und} \quad (9.13)$$

$$\hat{\Phi}_j = \text{res}(\hat{H}(s), \hat{\lambda}_j) = \hat{H}(s - \hat{\lambda}_j)|_{s=\hat{\lambda}_j}, \quad j = 1, \dots, r. \quad (9.14)$$

Mit dem folgenden Satz, der in [87] zu finden ist, und eine Verallgemeinerung der Resultate aus [84, 85] darstellt, lässt sich der Fehler des reduzierten Systems bezüglich der \mathcal{H}_2 -Norm bestimmen.

Satz 9.2 (Approximationsfehler bezüglich der \mathcal{H}_2 -Norm) *Gegeben seien die Übertragungsfunktionen eines unreduzierten SISO-Systems erster Ordnung $H(s)$ und eines reduzierten Systems $\hat{H}(s)$ der Dimension r . Seien λ_i und $\hat{\lambda}_i$ die Pole von $H(s)$ bzw. $\hat{H}(s)$ mit der zusätzlichen Annahme, dass die Pole von $\hat{H}(s)$ einfach sind. Seien Φ_i und $\hat{\Phi}_j$ die Residuen der Übertragungsfunktionen $H(s)$ bzw. $\hat{H}(s)$ an den Stellen ihrer Pole λ_i (9.13) bzw. $\hat{\lambda}_i$ (9.14). Dann ist die \mathcal{H}_2 -Norm des Fehlersystems $H_e(s) = H(s) - \hat{H}(s)$ durch*

$$\|H_e\|_{\mathcal{H}_2}^2 = \sum_{i=1}^{2n} \Phi_i \left(H(-\lambda_i) - \hat{H}(-\lambda_i) \right) - \sum_{j=1}^r \hat{\Phi}_j \left(H(-\hat{\lambda}_j) - \hat{H}(-\hat{\lambda}_j) \right) \quad (9.15)$$

gegeben.

BEWEIS: Siehe [87]. □

Diese Methode hat den Vorteil, eine globale Fehlerbestimmung in der \mathcal{H}_2 -Norm zu ermöglichen. Allerdings ist die Anwendung dieser Methode für große Modelle praktisch kaum anwendbar, da die $2n$ Eigenwerte des Originalsystems bekannt sein müssen.

9.2 Heuristische Methoden

In diesem Abschnitt werden existierende Methoden zur Abschätzung des Approximationsfehlers eines reduzierten Systems mit heuristischen Methoden vorgestellt.

Im ersten Abschnitt erfolgt eine Beschreibung der Methoden zur lokalen Fehlerbestimmung.

9.2.1 Lokale Fehlerschätzer

In diesem Abschnitt werden heuristische Methoden vorgestellt, die den Approximationsfehler Krylov-Unterraum reduzierter Modelle lokal bestimmen.

9.2.1.1 Heuristische Methoden

Die hier beschriebene heuristische Methode dient der lokalen Fehlerschätzung eines reduzierten Modells an einer Stelle des Frequenz- oder des Zeitbereiches. Eine ausführliche Darstellung der Methode ist in [21, 22] zu finden.

Als Fehlerschätzung dient die Differenz zwischen zwei reduzierten Systemen zunehmender Dimension. Dazu werden die reduzierten Systeme mit der Dimension r und $r_{neu} = r + \ell$ betrachtet, wobei $\ell \in \mathbb{N}$ die Zunahme der reduzierten Dimension nach einer Iteration des Reduktionsverfahrens bezeichnet.

Im Frequenzbereich lässt sich der relative Fehler an der Stelle s_0 über die Gleichung

$$\epsilon_{rel}(s_0, r) = \frac{\|H(s_0) - \hat{H}_r(s_0)\|}{\|H(s_0)\|} \quad (9.16)$$

bestimmen, wobei $\mathbf{H}(s_0)$ und $\hat{\mathbf{H}}_r(s_0)$ die Übertragungsfunktionen des originalen bzw. des reduzierten Systems der Dimension r an der Entwicklungsstelle s_0 und $\|\cdot\|$ eine geeignete Norm darstellen. Wird das reduzierte Modell sukzessive berechnet, wobei sich die Dimension des reduzierten Systems nach jeder Iteration um ℓ erhöht, dann läßt sich ein Schätzer des relativen Fehlers mit Hilfe des relativen Fehlers zweier aufeinander folgender reduzierter Systeme r und $r_{neu} = r + \ell$ bestimmen:

$$\hat{\epsilon}_{rel}(s_0, r_{neu}) = \frac{\|\hat{\mathbf{H}}_r(s_0) - \hat{\mathbf{H}}_{r_{neu}}(s_0)\|}{\|\hat{\mathbf{H}}_r(s_0)\|}. \quad (9.17)$$

Dieser Ansatz ist auch auf den Zeitbereich anwendbar. Dazu betrachtet man den relativen quadratischen Fehler im Zeitbereich:

$$\epsilon(T, r) = \sqrt{\frac{\sum_{t=0}^{T=N \cdot \Delta t} \left(\frac{\mathbf{Y}(t) - \hat{\mathbf{Y}}_r(t)}{\mathbf{Y}(t)} \right)^2}{N}}, \quad (9.18)$$

wobei $\mathbf{Y}(t)$ und $\hat{\mathbf{Y}}_r(t)$ den Ausgang des Originalsystems bzw. des reduzierten Systems der Dimension r in N diskreten Zeitpunkten t und einer Schrittweite von Δt darstellen.

Hier erhält man als Schätzer des absoluten Fehlers des reduzierten Systems:

$$\hat{\epsilon}(T, r) = \sqrt{\frac{\sum_{t=0}^{T=N \cdot \Delta t} \left(\frac{\hat{\mathbf{Y}}_r(t) - \hat{\mathbf{Y}}_{r+\ell}(t)}{\hat{\mathbf{Y}}_r(t)} \right)^2}{N}}. \quad (9.19)$$

Die Berechnung des Fehlerschätzers im Zeitbereich ist aufwendiger als im Frequenzbereich. Die Funktionen (9.18) und (9.19) sind keine Funktionen, die von der Zeit abhängen, sondern von der Dimension r des reduzierten Systems. Dabei muss der Fehler für jeden Wert von r über das gesamte relevante Zeitintervall hinweg ausgewertet werden.

Der Vorteil dieser heuristischen Methode liegt im relativ geringen Aufwand der Bestimmung des Fehlerschätzers. Dem steht die nur lokale Gültigkeit der Fehlerschätzung gegenüber. In der Nähe einer Entwicklungsstelle konvergiert der Fehler gegen den exakten Approximationsfehler, der auch mit reduzierten Modellen höherer Dimension nicht weiter verringert werden kann. Wird der Fehler hingegen an einer Stelle ausgewertet, der sich nicht in der Nähe einer Entwicklungsstelle befindet, oszilliert der Fehlerverlauf in Abhängigkeit von der Dimension des reduzierten Systems.

9.2.1.2 Fehlerschätzer nach Grimme

Auch der Fehlerschätzer

$$\hat{\epsilon}_{rel}(s_0, \hat{j}) = \sum_{j=1}^{\hat{j}} 2^{j-\hat{j}} \frac{\|\hat{\mathbf{H}}_j(s_0) - \hat{\mathbf{H}}_{j-1}(s_0)\|}{\|\hat{\mathbf{H}}_j(s_0)\|}. \quad (9.20)$$

basiert auf der Berechnung der Differenzen aufeinander folgender reduzierter Systeme $\hat{\mathbf{H}}_{j-1}(s_0)$ und $\hat{\mathbf{H}}_j(s_0)$ an der Stelle s_0 , wobei die relativen Differenzen zwischen den Übertragungsfunktionen nach \hat{j} Iterationen eines Krylov-Unterraum-Verfahrens mit Hilfe des Faktors $2^{j-\hat{j}}$ gewichtet werden. Dabei stellen $\hat{\mathbf{H}}_{j-1}$ und $\hat{\mathbf{H}}_j$ die Übertragungsfunktionen der nach der $j-1$ -ten bzw.

j -ten Iteration des Krylov-Unterraum-Verfahrens reduzierten Systeme und $\|\cdot\|$ eine geeignete Norm dar. Für eine ausführlichere Darstellung sei auf [80] verwiesen. Diese Methode der Fehlerschätzung hat den Vorteil, dass dazu nur die Fehler aufeinander folgender reduzierter Systeme, die eine geringe Dimension aufweisen, berechnet werden müssen.

9.2.2 Ein globaler Fehlerschätzer

Liegen die Hankel-Singulärwerte des originalen und eines reduzierten Systems in der Form (3.12) vor, ist mit der Abschätzung

$$\|\mathbf{H} - \hat{\mathbf{H}}_r\|_{\mathcal{H}_\infty} \leq 2(\sigma_{r+1} + \cdots + \sigma_{2n}) \quad (9.21)$$

eine globale Fehlerschranke gegeben [10]. Hierbei stellt \mathbf{H} und $\hat{\mathbf{H}}_r$ die Übertragungsfunktion des originalen bzw. des reduzierten Systems der Dimension r und σ_j den j -ten Hankel-Singulärwert der geordneten Menge $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{2n}$ des originalen Systems dar.

Da die Bestimmung der Hankel-Singulärwerte, wegen des großen Aufwandes bei großen Systemen unpraktikabel ist, wird in [21, 22] eine Methode vorgeschlagen, die nur auf der Bestimmung der Hankel-Singulärwerte reduzierter Systeme basiert.

Liegen die reduzierten Modelle als Systeme der Form (3.12) vor, erhält man nach i Iterationen, wobei sich die Dimension des reduzierten Systems nach jeder Iteration um ℓ erhöht, eine Matrix-Struktur der Hankel-Singulärwerte:

$$\Sigma_i = \begin{bmatrix} \hat{\sigma}_{1,1} & 0 & \cdots & 0 \\ \hat{\sigma}_{2,1} & \hat{\sigma}_{2,2} & \ddots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \hat{\sigma}_{i,1} & \hat{\sigma}_{i,2} & \cdots & \hat{\sigma}_{i,i\ell} \end{bmatrix}, \quad (9.22)$$

wobei $\hat{\sigma}_{i,j}$ der j -te Hankel-Singulärwert nach der i -ten Iteration des reduzierten Systems der Dimension $r = i \cdot \ell$ ist.

Nach einigen Iterationen konvergieren die größten Hankel-Singulärwerte des reduzierten Systems gegen die Hankel-Singulärwerte des Originalsystems. Nach einer geringen Anzahl von Iterationen werden somit die größten Hankel-Singulärwerte des Originalsystems gut approximiert. Sind nach der i -ten Iteration auf diese Weise die ersten k Hankel-Singulärwerte des Originalsystems $\hat{\sigma}_{i,j}$, $j = 1, \dots, k$ approximativ bestimmt und wird davon ausgegangen, dass alle Hankel-Singulärwerte $\sigma_{i,j}$, $j = k+1, \dots, 2n$ des Originalsystems den gleichen Wert wie der Hankel-Singulärwert $\hat{\sigma}_{i,k}$ aufweisen, dann kann die Gleichung (9.21) herangezogen werden, um den globalen Fehler abzuschätzen:

$$\|\mathbf{H} - \hat{\mathbf{H}}_r\|_{\mathcal{H}_\infty} \leq 2(\sigma_{i,k+1} + \cdots + \sigma_{i,2n}) \leq (2n - k) \cdot \hat{\sigma}_{i,k}. \quad (9.23)$$

Da mit der Annahme $\sigma_{i,j} = \hat{\sigma}_{i,k}$ für $j = k+1, \dots, 2n$ die bis hierher beschriebene Methode zu einer sehr groben Schätzung führt, muss die rechte Seite der Gleichung (9.23) durch eine Abschätzung des Abfalls der Hankel-Singulärwerte $\sigma_{i,j}$ für $j = k+1, \dots, 2n$ verbessert werden.

Dazu wird in [21] zum Abschätzen der oberen Schranken für die Hankel-Singulärwerte $\hat{\sigma}_{i,j}$ mit $j = k + 1, \dots, 2n$ die Abschätzung

$$\tilde{\sigma}_{i,j} \leq \left(\prod_{s=0}^{j-1} \frac{\kappa(\check{\mathbf{A}})^{(2s+1)/(2j) - 1}}{\kappa(\check{\mathbf{A}})^{(2s+1)/(2j) + 1}} \right)^2 \cdot \hat{\sigma}_{1,1} \quad (9.24)$$

angegeben, wobei $\kappa(\check{\mathbf{A}})$ die Konditionszahl einer symmetrischen Systemmatrix $\check{\mathbf{A}}$ ist. Die Werte $\tilde{\sigma}_{i,j}$, $j = k + 1, \dots, 2n$ können damit als Schätzer für die Hankel-Singulärwerte $\sigma_{i,j}$, $j = k + 1, \dots, 2n$ in der Gleichung (9.23) angesehen werden.

Der Vorteil dieser Methode liegt im relativ geringen Aufwand der Fehlerbestimmung, da nur die Hankel-Singulärwerte reduzierter Systeme bestimmt werden müssen. Da jedoch der ermittelte Fehler ohne Abschätzungen der oberen Schranken für die Hankel-Singulärwerte $\sigma_{i,j}$, $j = k + 1, \dots, 2n$ sehr grob sein kann und die Abschätzungen dieser Hankel-Singulärwerte nur für symmetrische Systeme effizient bestimmt werden können, beschränkt sich die Anwendbarkeit dieser Methode ausschließlich auf symmetrische Systeme.

9.3 Adaptive Methoden

Wie im Abschnitt (8.4) bereits dargestellt wurde, tritt bei der praktischen Anwendung von Krylov-Unterraum-Verfahren die Frage nach der Wahl der Entwicklungsstellen und der Stufen der dazu korrespondierenden Krylov-Unterräume auf. Gewonnene Informationen über den Verlauf des Approximationsfehlers des reduzierten Systems können verwendet werden, um aus einer Menge von gegebenen Stellen die optimalen Entwicklungsstellen auszuwählen [64, 81, 80] oder die Stufen der dazu korrespondierenden Krylov-Unterräume zu bestimmen [71, 116].

In [105, 154] werden Methoden, die den Reduktionsprozess automatisch abbrechen, vorgestellt. Diese basieren darauf, dass die Konvergenz der Hankel-Singulärwerte einer Folge reduzierter Systeme mit zunehmender Dimension als Abbruchkriterium verwendet wird.

Im folgenden werden zwei Verfahren, die adaptive Methoden zur Bestimmung der Stufen der verwendeten Krylov-Unterräume zu einer gegebenen Menge von Entwicklungsstellen verwenden, näher vorgestellt.

9.3.1 Adaptive Auswahl von Entwicklungsstellen

In [80, 81] wird eine Methode zur optimalen Auswahl der Entwicklungsstellen aus einer Menge fest vorgegebener Stellen s_i , $i = 1, \dots, \hat{i}$ für ein Verfahren zur Reduktion von SISO-Systemen erster Ordnung mit dem rationalem Lanczos-Verfahren vorgeschlagen.

Algorithmus 12 gibt den Ablauf dieser Methode zur adaptiven Auswahl fest vorgegebener Entwicklungsstellen wieder. Der dargestellte Algorithmus basiert dabei auf Algorithmus 2 in [80], der hier für Krylov-Unterraum-Verfahren zur Reduktion von SISO-Systemen erster Ordnung, die auf dem Arnoldi- oder dem Lanczos-Algorithmus basieren, angepasst dargestellt wird.

In Zeile 11 des Algorithmus 12 wird der (relative) Approximationsfehler

$$\hat{\epsilon}_{rel,k} = \frac{\|\hat{H}_k - \hat{H}_{k-1}\|}{\|\hat{H}_k\|} \quad (9.25)$$

Algorithmus 12 Krylov-Unterraum-Verfahren mit adaptiver Entwicklungsstellenauswahl nach Algorithmus 2 in [80]

Eingabe: Systemmatrizen $\mathbf{E}, \mathbf{A}, \mathbf{b}, \mathbf{c}$; fest vorgegebene Entwicklungsstellen $s_i, i = 1, \dots, \hat{i}; \epsilon_{min} < 1; \alpha$

Ausgabe: Reduzierte Systemmatrizen $\hat{\mathbf{E}}, \hat{\mathbf{A}}, \hat{\mathbf{b}}, \hat{\mathbf{c}}$

```

1: function  $[\hat{\mathbf{E}}, \hat{\mathbf{A}}, \hat{\mathbf{b}}, \hat{\mathbf{c}}] = \text{Adapt\_Krylov}(\mathbf{E}, \mathbf{A}, \mathbf{b}, \mathbf{c}; \epsilon_{min}; \alpha)$ 
2:  $\hat{\epsilon}_1 = 1; \sigma_1 = \sigma_2 = s_1; k = j = 1$ 
3: Berechne  $\mathbf{V}$  und  $\mathbf{W}$  mit einem Krylov-Unterraum-Verfahren und der Entwicklungsstelle  $\sigma_1$ 
4: Berechne  $\hat{H}_k$  mit den durch  $\mathbf{V}$  und  $\mathbf{W}$  reduzierten Systemmatrizen  $\hat{\mathbf{E}}, \hat{\mathbf{A}}, \hat{\mathbf{b}}$ 
5: while  $\hat{\epsilon}_k > \epsilon_{min}$  ▷ Solange der Fehler größer als  $\epsilon_{min}$ 
6:    $k = k + 1$ 
7:   Berechne  $\mathbf{V}_k$  und  $\mathbf{W}_k$  mit einem Krylov-Unterraum-Verfahren und Entwicklungsstelle  $\sigma_k$  so modifiziert,
     dass  $\mathbf{V}_k$  gegen  $\mathbf{V}$  und  $\mathbf{W}_k$  gegen  $\mathbf{W}$  orthogonalisiert wird
8:    $\mathbf{V} = [\mathbf{V} \ \mathbf{V}_k], \mathbf{W} = [\mathbf{W} \ \mathbf{W}_k]$ 
9:   Berechne  $\hat{H}_k$  mit den durch  $\mathbf{V}$  und  $\mathbf{W}$  reduzierten Systemmatrizen  $\hat{\mathbf{E}}, \hat{\mathbf{A}}, \hat{\mathbf{b}}$ 
10:   $\hat{\epsilon}_k = 0.5 (\hat{\epsilon}_{k-1} + \|\hat{H}_k - \hat{H}_{k-1}\| / \|\hat{H}_k\|)$  ▷ Fehlerberechnung
11:  if  $\|\hat{H}_k - \hat{H}_{k-1}\| / \|\hat{H}_k\| < (1 + \hat{\epsilon}_k \alpha j) \cdot \epsilon_{min}$  ▷ Stagnation des Fehlers
12:     $\sigma_{k+1} = \arg \max_{s_i, i=1, \dots, \hat{i}} \{r_b(s_i)\}$  ▷ neue Entwicklungsstelle
13:     $j = 0$ 
14:  else
15:     $\sigma_{k+1} = \sigma_k$ 
16:  end if
17:   $j = j + 1$ 
18: end while
19: Berechne mit  $\mathbf{V}$  und  $\mathbf{W}$  die reduzierten Systemmatrizen  $\hat{\mathbf{E}}, \hat{\mathbf{A}}, \hat{\mathbf{b}}, \hat{\mathbf{c}}$ 

```

zwischen dem reduzierten System der k -ten und der $(k-1)$ -ten Iteration als Kriterium zur Wahl der Stufe des zur Entwicklungsstelle σ_k korrespondierenden Krylov-Unterraumes verwendet. Dazu wird der Fehler an \hat{i} Stellen $s_i, i = 1, \dots, \hat{i}$ mit einer geeigneten Norm $\|\cdot\|$ berechnet. Zur Abschätzung des Fehlers wird in [80] der Fehlerschätzer (9.20) vorgeschlagen.

Eine Stelle s_i wird solange als Entwicklungsstelle σ_k in der k -ten Iteration ausgewählt, bis der Fehler $\hat{\epsilon}_k$ stagniert. In Zeile 11 des Algorithmus wird dazu die Stagnation des Fehlers beobachtet. Der Fehler $\hat{\epsilon}_{rel,k}$ stagniert an der Entwicklungsstelle σ_k , wenn gilt:

$$\hat{\epsilon}_{rel,k} < (1 + \hat{\epsilon}_k \alpha j) \cdot \epsilon_{min}, \quad (9.26)$$

wobei j die Anzahl der Iterationen mit der aktuellen Entwicklungsstelle, k die Anzahl der Iterationen des Algorithmus und α eine nichtnegative ganze Zahl darstellen.

In Zeile 12 des Algorithmus wird, falls der Fehler stagnierte, eine neue Entwicklungsstelle bestimmt. Dazu wird das Residuum $r_b(\sigma_i)$ für die \hat{i} potenziellen Entwicklungsstellen $\sigma_i, i = 1, \dots, \hat{i}$ berechnet. Die Berechnung von $r_b(\sigma_i)$ kann dabei mit der Gleichung (9.3) erfolgen. Die Entwicklungsstelle, die mit dem größtem Residuum korrespondiert, wird dann als neue Entwicklungsstelle verwendet.

Um die Rolle des Parameters α in der Gleichung (9.26) besser zu verstehen, können folgende Betrachtungen angestellt werden. Man erwartet eine schnelle Stagnation des Fehlers, wenn α groß gewählt wird. Der Nachteil eines groß gewählten α ist jedoch in der hohen Anzahl von verwendeten Entwicklungsstellen zu sehen, welches zu einer langen Berechnungszeit führen kann.

Die Konvergenz des Verfahrens ist relativ unabhängig von der Wahl von α , allerdings wirkt sich α auf die Konvergenzrate aus.

Anmerkung 9.2 *Mit der Verwendung der Gleichungen (9.5), (9.6) und (9.7) anstatt der Gleichungen (9.2), (9.3) und (9.4) zur Bestimmung des Approximationsfehlers $\hat{\epsilon}_k$ ist diese Methode auch auf MIMO-Systemen anwendbar.*

Die Vorteile dieser adaptiven Methode zur Bestimmung der Entwicklungsstellen liegt in der relativ einfachen Berechnung des Residuums $r_b(s)$ und des Fehlers $\hat{\epsilon}_k$.

Die Nachteile dieser Methode liegen in der Einführung des Parameters α , dessen optimale Wahl ein offenes Problem darstellt. Zudem können a priori keine Aussagen über die Anzahl der benötigten Entwicklungsstellen und der Dimension r des reduzierten Systems getroffen werden.

9.3.2 Adaptiver rationaler Arnoldi-Algorithmus (AORA)

Wie bereits im Abschnitt 8.4.5 erläutert wurde, stellt neben der Wahl der Entwicklungsstellen auch die Wahl der geeigneten Stufe der Krylov-Unterräume zu jeder Entwicklungsstelle bei den Krylov-Unterraum-Verfahren ein Problem dar. In diesem Abschnitt wird ein Verfahren vorgestellt, das eine adaptive Methode verwendet, um die Stufen der benötigten Krylov-Unterräume zu bestimmen.

Das Verfahren wird *adaptiver rationaler Arnoldi-Algorithmus*¹ (AORA) genannt, basiert auf dem rationalen Arnoldi-Algorithmus zur Reduktion von SISO-Systemen und bestimmt zu jeder Entwicklungsstelle aus einer Menge fest vorgegebener Entwicklungsstellen s_i , $i = 1, \dots, \hat{i}$ und fest vorgegebener Dimension des reduzierten Systems r , die Stufe des Krylov-Unterraumes k_i zu jeder Entwicklungsstelle s_i . Das Verfahren berechnet ein reduziertes SISO-System der Dimension r , welches in r Iterationen erzeugt wird. Dazu werden in jeder Iteration die Momentenfehler (9.8) an den potenziellen Entwicklungsstellen s_i berechnet und die Stelle mit dem betragsmäßig größten Momentenfehler als Entwicklungsstelle in dieser Iteration verwendet. Dabei kann eine Stelle s_i mehrfach oder auch gar nicht ausgewählt werden. Die hier gegebene Darstellung des Verfahrens basiert auf [116]. Für eine ausführlichere Darstellung dieses Verfahrens sei auf diese Arbeit verwiesen.

Stimmen in der j -ten Iteration des Verfahrens bereits die ersten $j_i - 1$ Momente der Übertragungsfunktion an den Stellen s_i mit denen des Originalsystems überein, dann werden in dieser Iteration die j_i -ten Momentenfehler $\epsilon_{j_i}(s_i)$ der Übertragungsfunktion des reduzierten Systems, an allen Stellen s_i aus dem gegebenen Satz von potenziellen Entwicklungsstellen, berechnet.

Die grundlegende Idee des Verfahrens besteht nun in der Auswahl der Stelle s_i als Entwicklungsstelle in der j -ten Iteration, an der sich der betragsmäßig größte Momentenfehler $\epsilon_{j_i}(s_i)$ befindet. Dabei wird durch diese Auswahl der Entwicklungsstelle die größte Reduzierung des Approximationsfehlers erwartet.

Diese Auswahlmethode hat zur Folge, dass eine Stelle, an der die Übertragungsfunktion schwer zu approximieren ist und die dadurch in mehreren Iterationen den größten Momentenfehler aufweist, mehrfach als Entwicklungsstellen ausgewählt wird, wohingegen Stellen, an denen

¹In Englisch: *Adaptive Order Rational Arnoldi Algorithm*

die Übertragungsfunktion des reduzierten Systems nur kleine Momentenfehler besitzen, unter Umständen kein einziges mal als Entwicklungsstelle verwendet werden. Diese Auswahl der Entwicklungsstellen führt demnach im Ablauf des Reduktionsprozesses zu einer adaptiven Bestimmung der Stufen der Krylov-Unterräume, die zu den jeweiligen Entwicklungsstellen korrespondieren.

In [116] wird mit Hilfe des folgenden Satzes der exakte Fehler des k -ten Momentes der Übertragungsfunktion eines reduzierten Systems erster Ordnung an einer Entwicklungsstelle s_0 analytisch hergeleitet.

Satz 9.3 (Bestimmung des k -ten Momentenfehlers - reduziertes SISO-System)

Resultiert ein reduziertes SISO-System erster Ordnung aus der Anwendung der Projektion (5.10) mit $\mathbf{W} = \mathbf{V}$, wobei die ersten $j = 0, 1, \dots, k-1$ Momente des reduzierten Systems mit den ersten Momenten des Originalsystems an der Entwicklungsstelle s_0 zur Übereinstimmung gebracht werden und die Matrix $\mathbf{V} = [\mathbf{v}_1 \mathbf{v}_2 \dots \mathbf{v}_k]$ mit Hilfe des Arnoldi-Algorithmus (Algorithmus 5) generiert wurde, dann gilt für den k -ten Momentenfehler $\varepsilon_k(s_0)$ an der Entwicklungsstelle s_0 :

$$\varepsilon_k(s_0) := H_k(s_0) - \hat{H}_k(s_0) = \mathbf{c} \left(\prod_{j=0}^k h_{j+1,j} \right) \mathbf{v}_{k+1}, \quad (9.27)$$

wobei $H_k(s_0)$ und $\hat{H}_k(s_0)$ die k -ten Momente der Übertragungsfunktionen des unreduzierten bzw. des reduzierten Systems bezeichnen und $h_{1,0} = \|(s_0 \mathbf{E} - \mathbf{A})^{-1} \mathbf{b}\|$ ist. Die Werte $h_{j+1,j}$ für $j = 1, \dots, k$ werden dabei durch den Arnoldi-Algorithmus berechnet (Zeile 12, Algorithmus 3).

BEWEIS: Siehe [42]. □

Basierend auf Satz 9.3 wird in [42] der folgende Satz bewiesen, dessen Aussage zur effizienten Bestimmung der j_i -ten Momentenfehler $\epsilon_{j_i}(s_i)$ an den Entwicklungsstellen s_i , $i = 1, \dots, \hat{i}$ im Ablauf des Verfahrens AORA verwendet wird.

Satz 9.4 (Berechnung der j_i -ten Momentenfehler beim Verfahren AORA)

Angenommen die ersten j_i Momente eines SISO-Systems und eines reduzierten SISO-Systems der Dimension $j-1 = \sum_{i=1}^{\hat{i}} j_i$ stimmen an den Entwicklungsstellen s_i überein, d.h. es gilt $H_\ell(s_i) = \hat{H}_\ell(s_i)$ mit $\ell = 0, 1, \dots, j_i-1$ für $i = 1, \dots, \hat{i}$. Die Systemmatrizen des reduzierten Systems seien dabei durch die Projektion (5.10) mit der Matrix \mathbf{V} (und $\mathbf{W} = \mathbf{V}$) generiert worden, die mit dem AORA-Algorithmus (Algorithmus 13) berechnet wurde. Der j_i -te Momentenfehler $\epsilon_{j_i}(s_i)$ der Übertragungsfunktion $\hat{H}_{j_i}(s_i)$ des reduzierten Systems nach der j -ten Iteration des Algorithmus an der Entwicklungsstelle s_i , kann dann über die Gleichung

$$\epsilon_{j_i}(s_i) = |H_{j_i}(s_i) - \hat{H}_{j_i}(s_i)| = |h_\pi(s_i) \mathbf{c} \mathbf{r}^{(j-1)}(s_i)| \quad (9.28)$$

ausgedrückt werden. Dabei ist $h_\pi(s_i) = \prod_{t=1}^{j-1} \Gamma_t$, mit

$$\Gamma_t = \begin{cases} \|\mathbf{r}^{(t)}(s_i)\|, & \text{falls } s_i \text{ Entwicklungsstelle in der } t\text{-ten Iteration} \\ 1, & \text{sonst} \end{cases} \quad (9.29)$$

und $\mathbf{r}^{(j-1)}(s_i)$ der Wert, der in Zeile 19 des Algorithmus 13 in der vorhergehenden $(j-1)$ -ten Iteration berechnet wurde.

BEWEIS: Siehe [116]. □

Diese Berechnung des (exakten) Momentenfehlers wird verwendet, um in Zeile 7 des Algorithmus des Verfahrens AORA (Algorithmus 13) in der j -ten Iteration die Entwicklungsstelle mit dem maximalen Momentenfehler

$$\sigma_j = \arg \max_{s_i} |h_\pi(s_i) \mathbf{c} \mathbf{r}^{(j-1)}(s_i)| \quad (9.30)$$

zu bestimmen. Der Wert $h_\pi(s_i)$ wird in Zeile 4 mit 1 initialisiert und in jeder Iteration, in der die Stelle s_i als Entwicklungsstelle ausgewählt wird, mit

$$h_\pi(s_i) = h_\pi(s_i) \cdot \|\mathbf{r}^{(j-1)}(s_i)\|_2 \quad (9.31)$$

aktualisiert (Zeile 10).

Die Vorteile des Verfahrens liegen in den verbesserten Approximationseigenschaften gegenüber dem rationalen Arnoldi-Algorithmus insbesondere bei einem gut gewählten Satz von Entwicklungsstellen. Als problematisch ist die Abhängigkeit der Approximationseigenschaften von der Wahl der Entwicklungsstellen und der Größe der Dimension des reduzierten Systems r anzusehen.

Algorithmus 13 Adaptiver rationaler Arnoldi-Algorithmus (AORA) nach [116]

Eingabe: Systemmatrizen $\mathbf{E}, \mathbf{A}, \mathbf{b}, \mathbf{c}$; Entwicklungsstellen $s_i, i = 1, \dots, \hat{i}$; reduzierte Dimension r

Ausgabe: Reduzierte Systemmatrizen $\hat{\mathbf{E}}, \hat{\mathbf{A}}, \hat{\mathbf{b}}, \hat{\mathbf{c}}$

```

1: function  $[\hat{\mathbf{E}}, \hat{\mathbf{A}}, \hat{\mathbf{b}}, \hat{\mathbf{c}}] = \text{AORA}(\mathbf{E}, \mathbf{A}, \mathbf{b}, \mathbf{c}; s_i; r)$ 
2: for  $i = 1 : \hat{i}$ 
3:    $\mathbf{r}^{(0)}(s_i) := (s_i \mathbf{E} - \mathbf{A})^{-1} \mathbf{b}$  ▷ Initialisierung
4:    $h_\pi(s_i) := 1$ 
5: end for
6: for  $j = 1 : r$  ▷ AORA-Iterationen
7:    $\sigma_j = \arg \max_{s_i} |h_\pi(s_i) \mathbf{c} \mathbf{r}^{(j-1)}(s_i)|$  ▷ Wähle Stelle mit maximalen Momentenfehler
8:    $h_{j,j-1}(\sigma_j) := \|\mathbf{r}^{(j-1)}(\sigma_j)\|_2$  ▷ Generiere orthogonalen Vektor an der Stelle  $s_j$ 
9:    $\mathbf{v}_j := \mathbf{r}^{(j-1)}(\sigma_j) / h_{j,j-1}(\sigma_j)$ 
10:   $h_\pi(\sigma_j) := h_\pi(\sigma_j) \cdot h_{j,j-1}(\sigma_j)$ 
11:  for  $i = 1 : \hat{i}$  ▷ Berechne  $\mathbf{r}^{(j)}(s_i)$  für die nächste Iteration
12:    if  $(s_i == \sigma_j)$ 
13:       $\mathbf{r}^{(j)}(s_i) := (s_i \mathbf{E} - \mathbf{A})^{-1} \mathbf{E} \mathbf{v}_j$ 
14:    else
15:       $\mathbf{r}^{(j)}(s_i) := \mathbf{r}^{(j-1)}(s_i)$ 
16:    end if
17:    for  $t = 1 : j$ 
18:       $h_{t,j}(s_i) := \mathbf{v}_t^* \mathbf{r}^{(j)}(s_i)$ 
19:       $\mathbf{r}^{(j)}(s_i) := \mathbf{r}^{(j)}(s_i) - h_{t,j}(s_i) \mathbf{v}_t$ 
20:    end for
21:  end for
22: end for
23:  $\tilde{\mathbf{V}} = [\mathbf{v}_1 \mathbf{v}_2 \dots \mathbf{v}_r]$ 
24: if (ein  $s_i$  nicht reell) ▷ Generiere eine reelle Projektionsmatrix  $\mathbf{V}$ 
25:    $\mathbf{V}_{re} = \text{Re}(\tilde{\mathbf{V}}), \mathbf{V}_{im} = \text{Im}(\tilde{\mathbf{V}})$ 
26:   Berechne die reduzierte  $\mathbf{QR}$ -Zerlegung:  $\mathbf{V} \mathbf{R} = ([\mathbf{V}_{re} \mathbf{V}_{im}])$ 
27: end if
28: Berechne  $\hat{\mathbf{E}} = \mathbf{V}^* \mathbf{E} \mathbf{V}, \hat{\mathbf{A}} = \mathbf{V}^* \mathbf{A} \mathbf{V}, \hat{\mathbf{b}} = \mathbf{V}^* \mathbf{b}$  und  $\hat{\mathbf{c}} = \mathbf{c} \mathbf{V}$  ▷ Reduzierte Systemmatrizen

```

Kapitel 10

Modellreduktion von Werkzeugmaschinen

Wie bereits in Kapitel 4 dargestellt wurde, werden mechanische Strukturen als lineare Differenzialgleichungssysteme zweiter Ordnung beschrieben, die das dynamische Verhalten der betrachteten Struktur widerspiegeln. Da diese Systeme meist eine sehr hohe Dimension aufweisen, müssen diese vor ihrer Verwendung in Simulationen in der Dimension reduziert werden. Dazu haben sich in der Vergangenheit Verfahren, die auf der Methode der modalen Reduktion basieren, als geeignete Reduktionsverfahren etabliert. In diesem Kapitel werden Reduktionsverfahren vorgestellt, die als Alternative zu den etablierten Verfahren der modalen Reduktion angewendet werden können. Diese Verfahren basieren auf den Methoden, die in Kapitel 8 vorgestellt wurden und lassen sich direkt auf mechanische Strukturen anwenden.

Im ersten Abschnitt werden zunächst geeignete Strukturmodelle vorgestellt, die zur Untersuchung der Eigenschaften der vorgeschlagenen Reduktionsverfahren dienen. Nach dieser Darstellung in Abschnitt 10.1, werden in Abschnitt 10.2 die speziellen Anforderungen an die reduzierten Modelle diskutiert, die zur Simulation von Werkzeugmaschinen eingesetzt werden sollen.

Da die mechanischen Strukturen von Werkzeugmaschinen gewöhnlich durch sehr hochdimensionale Systeme beschrieben werden, stehen in dieser Arbeit die Krylov-Unterraummethoden, welche sich zur Reduktion von großen Systeme anbieten, im Fokus der Betrachtungen.

Nach Kenntnis des Autors existieren bislang keine \mathcal{H}_2 -optimalen Verfahren zur Reduktion von Systemen zweiter Ordnung. In Abschnitt 10.3 wird deshalb ein neues Verfahren vorgestellt, das als Modifikation der \mathcal{H}_2 -optimalen Verfahren IRKA und MIRIAM, die bereits in Abschnitt 8.5 beschrieben wurden, angesehen werden kann und *modifizierter iterativer rationaler Krylov-Algorithmus*¹ (MIRKA) genannt wird. Dieses Verfahren ist speziell auf die Reduktion von Modellen von Werkzeugmaschinen angepasst und präsentiert eine modifizierte iterative Methode zur Entwicklungsstellenwahl. Dabei ist es mit diesen Verfahren möglich sowohl Systeme erster Ordnung als auch Systeme zweiter Ordnung zu reduzieren.

Das Verfahren MIRKA wird hier verwendet, um die im Anwendungsgebiet der Strukturmechanik auftretenden Systeme zweiter Ordnung zu reduzieren. Dabei verwendet das Verfah-

¹In Englisch: Modified Iterative Rational Krylov Algorithm

ren den Arnoldi-Algorithmus zur Bestimmung einer geeigneten Basis des verwendeten Krylov-Unterraumes. Diese Ausführung des Verfahrens MIRKA wird hier *modifizierter iterativer rationaler Arnoldi-Algorithmus für Systeme zweiter Ordnung*² (MIRA_2o) genannt. Das vorgestellte Verfahren MIRA_2o ist speziell auf die Reduktion von Strukturmodellen von Werkzeugmaschinen angepasst, die in Form von einem ungedämpften oder proportional gedämpften Systemen zweiter Ordnung vorliegen und präsentiert eine iterative Methode zur Entwicklungsstellenwahl. Die Darstellung dieses Verfahrens baut auf den Darstellungen in den Veröffentlichungen [57] und [58] des Autors auf, in denen verschiedene Ausführungen dieses Verfahrens basierend auf dem Lanczos- und dem Arnoldi-Algorithmus untersucht werden.

In Abschnitt 10.4 wird ein neues Verfahren vorgestellt, das auf dem Verfahren MIRKA aufbaut und auf einer Kombination der Methode der modifizierten iterativen Entwicklungsstellenwahl mit dem globalen Arnoldi-Algorithmus basiert. Der globale Arnoldi-Algorithmus stellt eine Version des Block-Arnoldi-Algorithmus dar, der dazu verwendet werden kann, eine Basis des Block-Krylov-Unterraumes effizient zu berechnen. Dieses Verfahren ist direkt auf Systeme zweiter Ordnung anwendbar und zeichnet sich durch eine erhöhte Effizienz bei der Reduktion von MIMO-Systemen aus. Zudem dient dieser Algorithmus als Grundlage, der in den folgenden Abschnitten dargestellten automatisierten Verfahren.

Im darauf folgenden Abschnitt 10.4.4 wird ein Verfahren vorgestellt, das das vorhergehende Verfahren um eine adaptive Bestimmung der Stufe des Krylov-Unterraumes zu jeder Entwicklungsstelle ergänzt.

In Abschnitt 10.5 wird aufbauend auf den Verfahren der vorhergehenden Abschnitte eine Vorgehensweise zur automatischen Modellreduktion von mechanischen Strukturmodellen vorgeschlagen.

Die Approximationseigenschaften der mit den vorgestellten Verfahren reduzierten Modelle werden in den jeweiligen Abschnitten analysiert und diskutiert. Dabei werden die im Abschnitt 10.1.2 vorgestellten Strukturmodelle verwendet. Eine Diskussion der Ergebnisse bei Verwendung der reduzierten Systeme in einer Simulationsumgebung erfolgt im nachfolgenden Kapitel 11.

10.1 Anwendungsbeispiele

Die Approximationseigenschaften der mit den vorgestellten Verfahren reduzierten Systeme werden anhand von zwei Beispielen von FEM-Strukturmodellen verdeutlicht. Das erste Modell, einer abstrahierten Struktur einer Werkzeugmaschine mit 4983 Freiheitsgraden, dient dazu die grundsätzlichen Eigenschaften der neu entwickelten Verfahren zu untersuchen und mit bekannten Reduktionsverfahren zu vergleichen.

Beim zweiten Modell handelt es sich um das Modell einer Werkzeuginheit einer Werkzeugmaschine mit 54885 Freiheitsgraden. Diese Modelldimension ist bereits von einer Größe, die bezüglich des entstehenden numerischen Aufwandes für einige Reduktionsverfahren als problematisch anzusehen ist. Die Wahl eines Modells dieser Größe dient zur Verdeutlichung, dass die

²In Englisch: Modified Iterative Rational Arnoldi Algorithm

Eigenschaften der in dieser Arbeit vorgeschlagenen Verfahren auch auf hochdimensionale und komplex modellierte Strukturmodelle übertragbar sind.

Alle Berechnungen zur Modellreduktion und der Fehlerbestimmung in dieser Arbeit erfolgten auf einem Rechner mit Intel(R) Core(TM)2 CPU, 1.87 GHz und 2 GB RAM Arbeitsspeicher. Die Implementierung der verschiedenen Algorithmen der Reduktionsverfahren und die Auswertung der Reduktionsergebnisse erfolgte mit der Berechnungssoftware MATLAB/Simulink® in der Version 7.06, Release R2008a (64-bit-Version). Das Generieren der Originalsysteme und deren modal reduzierter Systeme erfolgte mit Hilfe des FEM-Programms MSC.Patran/Nastran®,³.

10.1.1 Fehlerbestimmung reduzierter Systeme

Um die in dieser Arbeit vorgestellten Reduktionsverfahren validieren zu können, werden Fehlermaße benötigt, die eine Aussage bezüglich der Approximationsgüte der mit diesen Verfahren reduzierten Systeme erlauben.

In diesem Abschnitt werden die in dieser Arbeit verwendeten Fehlermaße dargestellt. Dabei wird eine Fehlerbestimmung der reduzierten Systeme sowohl im Frequenzbereich als auch im Zeitbereich benötigt. Da hierzu im Gegensatz zu den im Kapitel 9 dargestellten Methoden die Bestimmung des Approximationsfehlers nicht im Zuge des Reduktionsprozesses durchgeführt wird, können dazu Methoden verwendet werden, die den Approximationsfehler möglichst exakt wiedergeben.

10.1.1.1 Fehlerbestimmung im Frequenzbereich

Um den Approximationsfehler der Übertragungsfunktion \hat{H} eines reduzierten Systems anzugeben, existieren mehrere Möglichkeiten. Als Kriterium zur Beurteilung des Fehlers sind die Größen

$$\epsilon_{abs,\infty} := \|\mathbf{H}(s) - \hat{\mathbf{H}}(s)\|_{\mathcal{H}_\infty} \quad \text{und} \quad \epsilon_{rel,\infty} := \frac{\|\mathbf{H}(s) - \hat{\mathbf{H}}(s)\|_{\mathcal{H}_\infty}}{\|\mathbf{H}(s)\|_{\mathcal{H}_\infty}} \quad (10.1)$$

oder

$$\epsilon_{abs,2} := \|\mathbf{H}(s) - \hat{\mathbf{H}}(s)\|_{\mathcal{H}_2} \quad \text{und} \quad \epsilon_{rel,2} := \frac{\|\mathbf{H}(s) - \hat{\mathbf{H}}(s)\|_{\mathcal{H}_2}}{\|\mathbf{H}(s)\|_{\mathcal{H}_2}} \quad (10.2)$$

üblich, wobei $\epsilon_{abs,\infty}$ und $\epsilon_{rel,\infty}$ absoluter bzw. relativer \mathcal{H}_∞ -Fehler und $\epsilon_{abs,2}$ und $\epsilon_{rel,2}$ absoluter bzw. relativer \mathcal{H}_2 -Fehler genannt wird, siehe z. B. [10]. Dabei handelt es sich bei $\|\cdot\|_{\mathcal{H}_\infty}$ und $\|\cdot\|_2$ um die Hardy-Normen \mathcal{H}_∞ bzw. \mathcal{H}_2 , die in Abschnitt 3.5 bereits näher vorgestellt wurden. Für stabile Systeme erster Ordnung ist es möglich, die Hardy-Normen mit Methoden der linearen Algebra zu berechnen. Die Verwendung dieser Normen bietet sich jedoch nur zur Bestimmung des Approximationsfehlers von Systemen kleiner Dimension an, da für Modelle mit großen Dimensionen die Berechnungen sehr rechenintensiv werden können. Aus diesem Grund erfolgt zur Bestimmung des Approximationsfehlers von hochdimensionalen Systemen eine punktweise Auswertung der Übertragungsfunktion. Dabei wird für eine Folge $\{f_k\}_{k=0}^{k_{max}}$ von linear oder logarithmisch verteilten Stellen innerhalb eines betrachteten Frequenzbereiches $[f_{min}, f_{max}]$ durch

³MSC.Software, Nastran, Kurzform für Nasa Structural Analysis System, von der US-Raumfahrtbehörde NASA entwickeltes Finite-Elemente-Programm.

Berechnen der entsprechenden Matrixgleichungen (3.24) bzw. (3.27) oder (3.28) die Werte der Übertragungsfunktion des Originalsystems $\mathbf{H}(2\pi\iota f_k)$ bzw. die Übertragungsfunktion des reduzierten Systems $\hat{\mathbf{H}}(2\pi\iota f_k)$ bestimmt.

Um Aussagen über die Approximationsgüte der reduzierten Systeme in einem, für den Anwender interessanten Frequenzintervall $[f_{min}, f_{max}]$ machen zu können, wurden folgende Fehlergrößen verwendet. Betrachtet man eine Menge von Indexmengen $\mathcal{I} \subseteq \{1, \dots, q\} \times \{1, \dots, m\}$, wobei m die Anzahl der Eingänge und q die Anzahl der Ausgänge bezeichnet, dann lassen sich nach [109] folgende Fehler definieren:

$$\epsilon_{rel}(f, \mathcal{I}) := \max \left\{ \frac{|[\mathbf{H}(2\pi\iota f)]_{ij} - [\hat{\mathbf{H}}(2\pi\iota f)]_{ij}|}{|[\mathbf{H}(2\pi\iota f)]_{ij}|} : (i, j) \in \mathcal{I} \right\} \quad (10.3)$$

und

$$\epsilon_{max}(\mathcal{I}) := \max \{ \epsilon_{rel}(f_k, \mathcal{I}) : k = 0, 1, \dots, k_{max} \}. \quad (10.4)$$

Mit $\epsilon_{max}(\mathcal{I})$ erhält man eine Art größten relativen Fehler des Frequenzganges für die mit \mathcal{I} selektierten Eingang-Ausgang-Paare. Mit diesem Fehlermaß lässt sich berücksichtigen, dass für manche Modelle der Approximationsfehler des Frequenzganges nur für bestimmte Eingang-Ausgang-Paare von Interesse ist. In der vorliegenden Arbeit wurden stets alle Eingang-Ausgang-Paare betrachtet.

Um die Approximationsgüte der reduzierten Modelle für alle Eingang-Ausgang-Paare beurteilen zu können, kann als weiteres Kriterium der Verlauf des relativen Approximationsfehlers an der Stelle f :

$$\epsilon_{rel}(f) := \frac{\|\mathbf{H}(2\pi\iota f) - \hat{\mathbf{H}}(2\pi\iota f)\|_F}{\|\mathbf{H}(2\pi\iota f)\|_F} \quad (10.5)$$

und

$$\epsilon_{max} := \max \{ \epsilon_{rel}(f_k) : k = 0, 1, \dots, k_{max} \}, \quad (10.6)$$

wobei \mathbf{H} und $\hat{\mathbf{H}}$ die Übertragungsfunktionen des originalen bzw. des reduzierten Systems und $\|\cdot\|_F$ die Frobenius-Norm darstellen, verwendet werden, siehe z. B. [117].

Die Betrachtung der Fehlergrößen (10.3) bis (10.6) zur Beurteilung der Approximationsgüte kann als problematisch angesehen werden. Da ϵ_{rel} nur in endlich vielen Punkten ausgewertet wird, bleibt die Größe des Fehlers außerhalb der lokalen Umgebung dieser Punkte unberücksichtigt. Es ist somit möglich, dass man keinen Unterschied zwischen verschiedenen reduzierten Systemen feststellt, da im untersuchten Frequenzbereich alle Verfahren nahezu konvergent sind. Dennoch können, vor allem bei hohen Frequenzen, außerhalb dieses Bereiches wesentliche Unterschiede bestehen.

Zieht man allerdings in Betracht, dass in der praktischen Anwendung von strukturmechanischen Modellen der Fehler nur in einem bestimmten niederen Frequenzbereich von Interesse ist, dann ist dieser Nachteil hinnehmbar.

Bei Modellen, deren Größe eine Berechnung der relativen \mathcal{H}_2 -Norm zulassen, wird zudem der integrale Ausdruck

$$\epsilon_{rel} := \frac{\left(\int_{f_{min}}^{f_{max}} \|\mathbf{H}(2\pi\iota f) - \hat{\mathbf{H}}(2\pi\iota f)\|_F^2 df \right)^{1/2}}{\left(\int_{f_{min}}^{f_{max}} \|\mathbf{H}(2\pi\iota f)\|_F^2 df \right)^{1/2}} \quad (10.7)$$

als skalarcs Maß für den relativen Approximationsfehler herangezogen [117]. Die Definition von ϵ_{rel} erfolgt dabei in Anlehnung an die \mathcal{H}_2 -Norm (3.137). Im Gegensatz zur \mathcal{H}_2 -Norm sind mit Hilfe dieser Größe jedoch Aussagen bezüglich eines Frequenzintervalls $[f_{min}, f_{max}]$ möglich, dessen Approximation für das in dieser Arbeit betrachtete Anwendungsgebiet von Interesse ist.

10.1.1.2 Fehlerbestimmung im Zeitbereich

Um Aussagen über die Approximationsgüte der reduzierten Systeme zu Zeitpunkten $t_k, k = 0, \dots, k_{max}$ innerhalb eines für den Anwender interessanten Zeitbereiches $[t_{min}, t_{max}]$ machen zu können, wird, analog zur Fehlerbestimmung im Frequenzbereich, die Menge $\mathcal{I} \subseteq \{1, \dots, q\} \times \{1, \dots, m\}$ von Indexmengen, wobei m die Anzahl der Eingänge und q die Anzahl der Ausgänge bezeichnet, betrachtet. Damit lässt sich der folgende Fehler definieren:

$$\delta_{abs}(t, \mathcal{I}) := \max \{ | [\mathbf{y}(t)]_{ij} - [\hat{\mathbf{y}}(t)]_{ij} |, : (i, j) \in \mathcal{I}, t \in [t_{min}, t_{max}] \} \quad \text{und} \quad (10.8)$$

$$\delta_{max} := \max \{ \delta_{abs}(t_k, \mathcal{I}) : k = 0, 1, \dots, k_{max} \}, \quad (10.9)$$

wobei $[\mathbf{y}(t)]_{ij}$ und $[\hat{\mathbf{y}}(t)]_{ij}$ die Lösung des Systems vom j -ten Eingang zum i -ten Ausgang des originalen bzw. des reduzierten Systems zum Zeitpunkt $t \in [t_{min}, t_{max}]$ darstellen.

10.1.2 Verwendete Beispielmodelle

Die Validierung der Reduktionsergebnisse, der in dieser Arbeit entwickelten Verfahren, erfolgt anhand von zwei Testmodellen. Bei beiden Testmodellen handelt es sich um abstrahierte Strukturen einer Werkzeugmaschine, die mit Hilfe von parametrischen Modellbausteinen, wie sie in Abschnitt 7.3.2.2 vorgestellt wurden, erstellt werden. Da die Modellierung nicht Teil der vorliegenden Arbeit ist, wird auf Modelle zurückgegriffen, die am Institut für Werkzeugmaschinen und Betriebswissenschaften⁴ (*iwb*) der Technischen Universität München erstellt wurden. Die Erstellung der Modelle erfolgte durch die Vorgehensweise, wie sie in Abschnitt 7.3 beschrieben wurde.

Die Massenmatrizen \mathbf{M} und die Steifigkeitsmatrizen \mathbf{K} der Modelle wurden dabei mit Hilfe des kommerziellen FEM-Programms MSC.Patran/Nastran[®] erzeugt. Nach Fertigstellung des FEM-Modells wurden die Koeffizientenmatrizen (Massen- und Steifigkeitsmatrix) der Bewegungsgleichungen vom FEM-Programm aufgestellt und mit Hilfe eines Makros ausgeleitet. Dabei wurden diejenigen Freiheitsgrade, welche bestimmten Zwangbedingungen unterliegen, sei es aufgrund von Einspannungen oder aufgrund linearer Abhängigkeiten durch bestimmte Elementtypen (Multi Point Constraints, MPCs), aus den Matrizen entfernt. Die Modelle liegen also als Massen- und Steifigkeitsmatrix mit allen unabhängigen, ansprechbaren Freiheitsgraden vor. Die Massenmatrix \mathbf{M} der Modelle ist dabei im Allgemeinen singulär und die Dämpfungsmatrix steht nicht direkt zur Verfügung, deshalb werden die ungedämpften Strukturen mit einer proportionalen Dämpfung beaufschlagt.

⁴Institut für Werkzeugmaschinen und Betriebswissenschaften, Technische Universität München, Boltzmannstr. 15, 85747 Garching

10.1.2.1 Erstes Beispielmodell: Werkzeugmaschine mit 4 983 Freiheitsgraden

In Abbildung 10.1(a) ist das FEM-Modell einer abstrahierten Struktur einer Werkzeugmaschine dargestellt, das als erstes Beispielmodell verwendet wurde. Dieses wurde zum Testen der Approximationseigenschaften der verschiedenen Reduktionsverfahren verwendet. Das entstandene Modell enthält nach Abzug aller linear abhängigen Freiheitsgrade, die aufgrund von Zwangsbedingungen nicht frei ansprechbar sind, 4 983 Freiheitsgrade.

Das Modell besteht aus den Hauptkomponenten Maschinenbett, Ständer, einem vereinfachten Support und der daran montierten Spindel, die in einer offenen C-Struktur aufgebaut sind, wie sie auch bei Bettfräsmaschinen vorkommt. Zusätzlich ist zwischen dem Bett und dem Ständer eine Vorschubachse mit Kugelgewindetrieb und Linearführungen modelliert, um in der Mechaniksimulation auch kleine Verfahrbewegungen des Fahrständers in y-Richtung realisieren zu können. Das Testmodell hat vier Eingänge (y-Rotation der Motorwelle, x-, y-, z-Translation des TCP) und sechs Ausgänge (y-Rotation Motorwelle, y-Rotation Spindelende, y-Translation am Führungsschuh, x-, y-, z-Verlagerung am TCP).

Die ungedämpfte Struktur wurde mit einer proportionalen Dämpfung mit $\alpha = 0.02$ und $\beta = \alpha/1500$ beaufschlagt. Das entstandene Modell enthält nach Abzug aller linear abhängigen Freiheitsgrade, die aufgrund von Zwangsbedingungen nicht frei sind, 4 983 Freiheitsgrade.

Die Abbildungen 10.1(b) und 10.1(c) stellen die Strukturen der Massen- bzw. der Steifigkeitsmatrix des Modells dar. Der Frequenzgang des MIMO-Systems kann der Abbildung 10.2 entnommen werden.

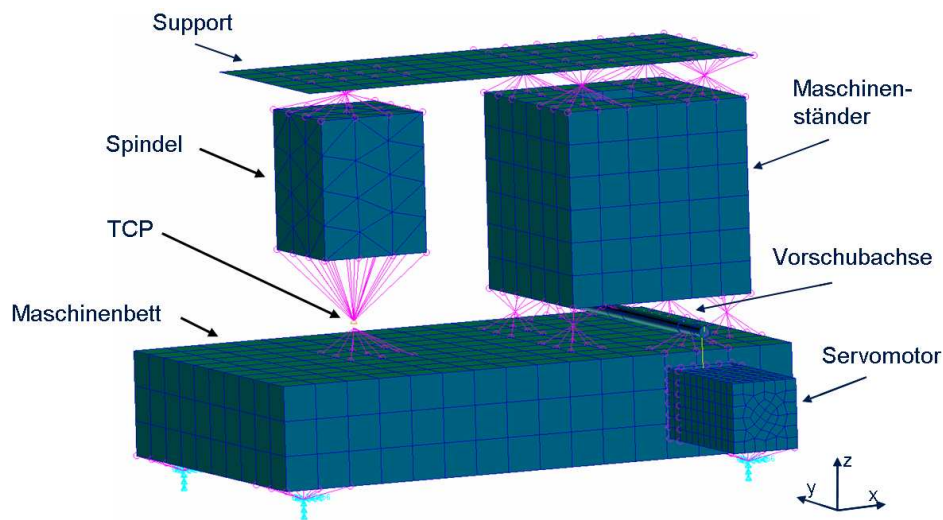
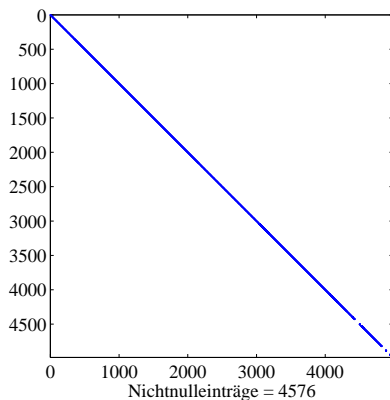
10.1.2.2 Zweites Beispielmodell: Werkzeugmaschine mit 54 885 Freiheitsgraden

Als zweites Beispiel dient das FEM-Modell einer Fräseinheit eines Bearbeitungszentrums einer modernen Werkzeugmaschine. Modelliert sind zwei Schlitten inklusive der Vorschubachsen, die für eine Vorschubbewegungen der Frässpindel benötigt werden. Das Testmodell hat vier Eingänge (y-Rotation der Motorwelle, x-, y-, z-Translation des TCP) und vier Ausgänge (y-Rotation Motorwelle, x-, y-, z-Verlagerung am TCP).

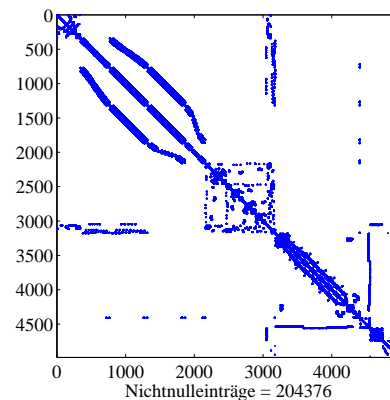
Die ungedämpfte Struktur wurde mit einer proportionalen Dämpfung mit $\alpha = 0.001$ und $\beta = \alpha/1500$ beaufschlagt. Das entstandene Modell enthält nach Abzug aller linear abhängigen Freiheitsgrade, die aufgrund von Zwangsbedingungen nicht frei ansprechbar sind, 54 885 Freiheitsgrade. Die Abbildung 10.3(a) stellt die modellierte Fräseinheit und die Abbildungen 10.3(b) und 10.3(c) stellen die Strukturen der Massen- bzw. der Steifigkeitsmatrix des Modells dar. Der Frequenzgang des MIMO-Systems kann der Abbildung 10.4 entnommen werden.

10.2 Spezielle Anforderungen an die Reduktionsverfahren

Soll ein Reduktionsverfahren auf die hier betrachteten Strukturmodelle angewendet werden, die zur Simulation des Verhaltens von Werkzeugmaschinen dienen, so muss dieses einige besondere Bedingungen erfüllen. Der Grund dafür ist in der Einbettung der reduzierten Modelle in Simulationsumgebungen zu sehen. Da zur Simulation von Werkzeugmaschinen üblicherwei-


 (a) FEM-Modell (Erstellt von T. Bonin, *iwb*, Technische Universität München.)


(b) Struktur der Massenmatrix



(c) Struktur der Steifigkeitsmatrix

Abbildung 10.1: FEM-Modell einer Werkzeugmaschine mit 4983 Freiheitsgraden - Erstes Beispielmodell

se kommerzielle Simulationsumgebungen zum Einsatz kommen, werden neben den generellen Anforderungen an reduzierte Systeme, wie sie in Abschnitt 5.1.2 bereits dargestellt wurden, weitere spezielle Anforderungen gestellt. Diese Anforderungen sollen im weiteren näher dargestellt werden. Um detaillierte Aussagen über das Verhalten der Strukturmodelle machen zu können, sind Vernetzungen mit einer hohen Anzahl an Gitterpunkten notwendig. Dies führt zu Modellen mit einer hohen Anzahl an Freiheitsgraden. Dies spiegelt sich wiederum in einer hohen Anzahl an Systemgleichungen wider. Die zu reduzierenden Systeme besitzen deshalb hochdimensionale aber dünn besetzte Systemmatrizen. Eine Bedingung an die Reduktionsverfahren ist somit die Berücksichtigung der dünn besetzten Struktur der Systemmatrizen. Die Implementierung der entwickelten Verfahren erfolgte in dieser Arbeit mit Rechenoperationen für dünn besetzte Matrizen und Vektoren. Dabei wurden die in MATLAB[®] vorhandenen Funktionen verwendet. Dünn besetzte Matrizen und Vektoren wurden in MATLAB[®] im Dateiformat für dünn besetzte Matrizen abgespeichert.

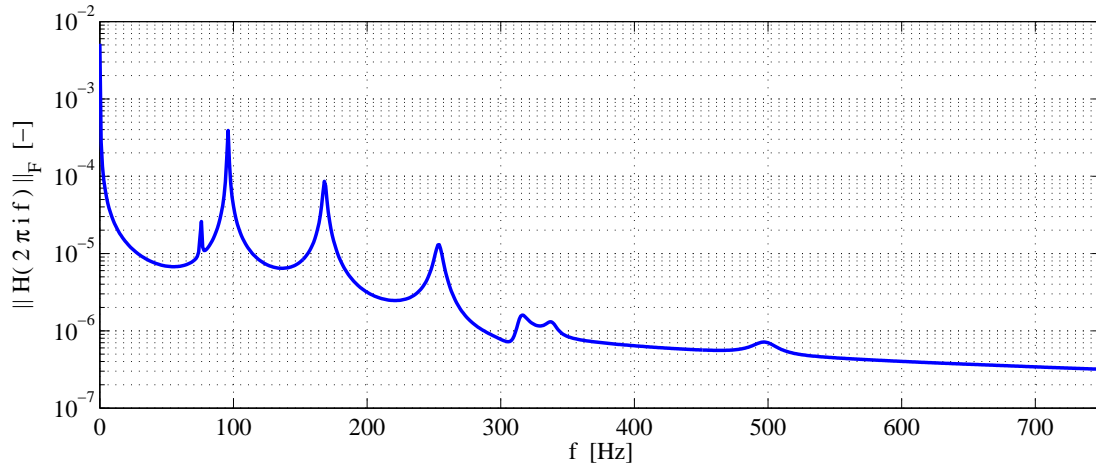


Abbildung 10.2: Frequenzgang des MIMO-Systems - Erstes Beispielmodell

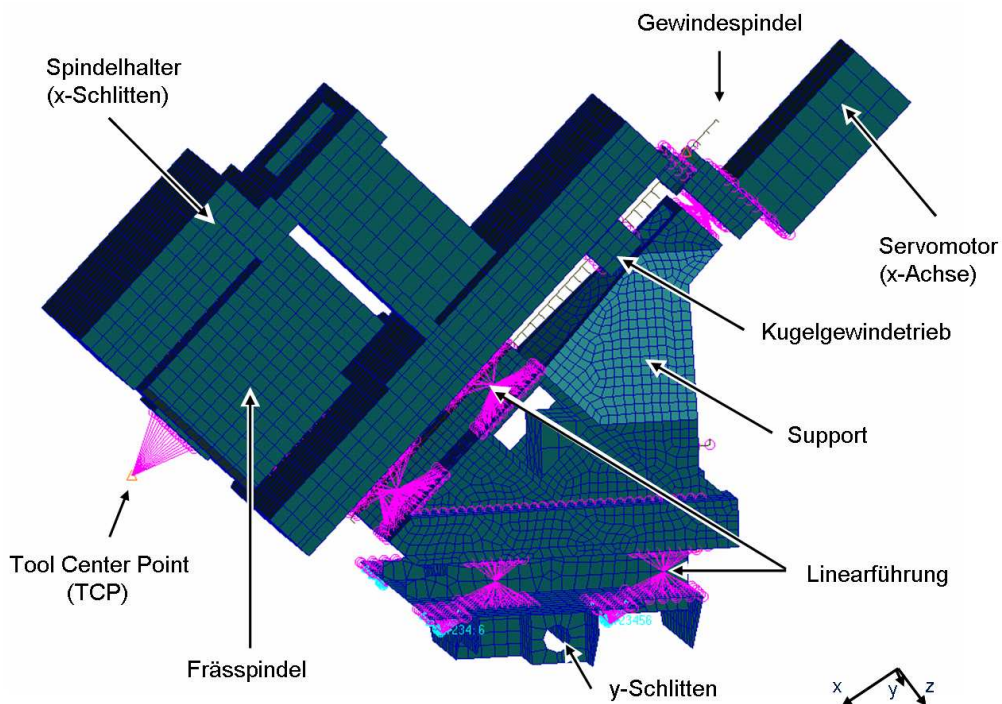
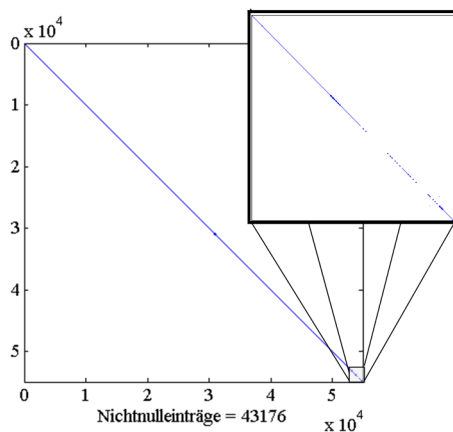
Die bei der Modellierung der mechanischen Struktur resultierenden Systeme besitzen üblicherweise mehrere Ein- und Ausgänge, deren Anzahl mit der Komplexität der modellierten mechanischen Struktur zunimmt. Für ein Reduktionsverfahren bedeutet dies, dass MIMO-Systeme möglichst effizient berechnet werden müssen.

Sollen reduzierte Systeme von Modellen mechanischer Strukturen für eine Regelsimulation mit einer Regelstrecke gekoppelt werden, liegen die zugrunde liegenden Originalsysteme der mechanischen Struktur als Systeme zweiter Ordnung mit Ausgangsmatrizen \mathbf{C}_p und \mathbf{C}_v vor. Diese Systeme weisen somit die Form (3.1) mit $\mathbf{C}_v \neq \mathbf{0}$ auf. Diese Darstellungsform soll nach einer Reduktion des Systems zweiter Ordnung erhalten bleiben.

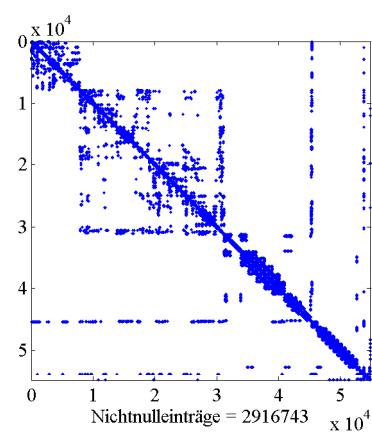
In der Praxis werden zudem zunehmend Verfahren verlangt, die die Reduktion automatisch durchführen. Darunter werden hier Verfahren bezeichnet, die vom Anwender nur die Eingaben eines Frequenzintervalls, das approximiert werden soll, und einer Fehlerschranke, die der Approximationsfehler des reduzierten Systems nicht überschreiten soll, benötigen. Weiter soll die Dimension des reduzierten Systems durch Vorgabe einer maximalen reduzierten Dimension begrenzt werden können. Bezogen auf Reduktionsverfahren basierend auf Krylov-Unterräumen, werden dazu Verfahren benötigt, die die Wahl der Entwicklungsstellen und die Stufen des zu jeder Entwicklungsstelle generierten Krylov-Unterraumes adaptiv bestimmen.

Wie in Abschnitt 8.4.5 bereits dargestellt wurde, existieren zwar Ansätze zur Wahl der Entwicklungsstellen und zur adaptiven Wahl der Stufe des zu jeder Entwicklungsstelle generierten Krylov-Unterraumes. Es existiert nach Kenntnis des Autors jedoch noch kein Verfahren, das eine Methode zur optimalen Wahl der Entwicklungsstellen mit einer adaptiven Bestimmung der Stufen der zugehörigen Krylov-Unterräume kombiniert. In dieser Arbeit wird deshalb durch Modifikationen und Erweiterungen bestehender adaptiver Methoden eine mögliche Automatisierung des Reduktionsprozesses vorgestellt.

Neben einem automatischen Reduktionsprozess werden weitere Anforderungen an das Reduktionsverfahren gestellt, so sollen zum Beispiel wichtige Struktureigenschaften, wie die Stabilität des Originalsystems, auf das reduzierte System übertragen werden. Zur Simulation von Werk-


 (a) FEM-Modell (Erstellt von T. Bonin, *iwb*, Technische Universität München.)


(b) Struktur der Massenmatrix



(c) Struktur der Steifigkeitsmatrix

Abbildung 10.3: FEM-Modell einer Werkzeugmaschine mit 54.885 Freiheitsgraden - Zweites Beispielmodell

zeugmaschinen werden üblicherweise kommerzielle Simulationsumgebungen verwendet, die ausschließlich mit reellwertigen Systemmatrizen operieren. Deshalb muss das Reduktionsverfahren reellwertige Projektionsmatrizen generieren können, um ein reellwertiges reduziertes System zu erhalten.

Bei den in dieser Arbeit entwickelten Reduktionsverfahren wird zudem sichergestellt, dass das reduzierte System stabil ist.

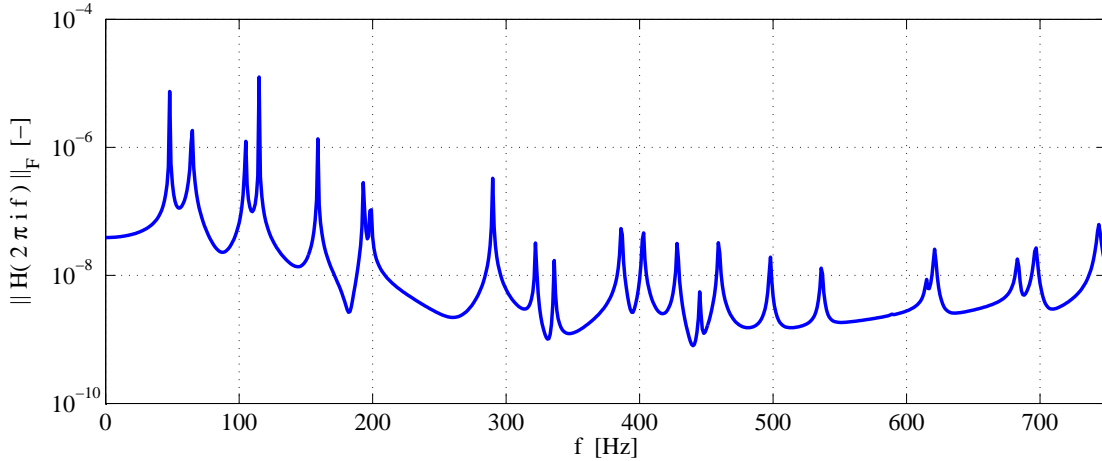


Abbildung 10.4: Frequenzgang des MIMO-Systems - Zweites Beispielmodell

10.3 Verfahren basierend auf dem Block-Arnoldi-Algorithmus

Bei der Reduktion von Strukturmodellen in Form von Systemen zweiter Ordnung werden vorwiegend einseitige Krylov-Unterraum-Verfahren verwendet, da diese die Eigenschaft der Passivitäts- und Stabilitäts-erhaltung aufweisen [155].

Da der Arnoldi-Algorithmus zudem einfacher zu implementieren und numerisch stabiler als der Lanczos-Algorithmus ist, stehen im weiteren Vorgehen dieser Arbeit einseitige Verfahren auf Basis des Rational-Arnoldi-Algorithmus im Vordergrund, um die in Form von MIMO-Systemen zweiter Ordnung vorliegenden Strukturmodelle passivitäts- und stabilitäts-erhaltend zu reduzieren.

Nach einer Darstellung des Block-Arnoldi-Algorithmus, auf dem die in diesem Abschnitt vorgestellten Reduktionsverfahren basieren, wird ein neues Verfahren vorgestellt, das für die Anwendung auf Strukturmodelle geeignete Entwicklungsstellen iterativ bestimmt.

10.3.1 Der Block-Arnoldi-Algorithmus

Die Implementierung von Verfahren für die Reduktion von MIMO-Systemen kann auf zwei unterschiedliche Arten erfolgen. Bei Verfahren der ersten Art, erfolgt die Berechnung der Basisvektoren des zugrundeliegenden Krylov-Unterraumes spaltenweise. Diese Art der Implementierung der Krylov-Unterraum-Verfahren wurde bereits mit den Algorithmen in Kapitel 8.4 dargestellt. Bei der zweiten Art der Implementierung erfolgt die Berechnung der Basisvektoren blockweise. Der Algorithmus 14 gibt diese Art der Implementierung in Pseudocode wieder. Der dargestellte Algorithmus basiert dabei auf der Blockversion des klassischen Arnoldi-Algorithmus, wie er in [153] dargestellt wird. Die Berechnung der Matrix \mathbf{W} in Zeile 7 des Algorithmus kann zu einer Matrix mit linear abhängigen Spaltenvektoren führen. Um dies zu vermeiden, wird der hier dargestellte Algorithmus um eine Rank-Revealing- \mathbf{QR} -Zerlegung und einer Elimination linear abhängiger Spaltenvektoren ergänzt (Zeilen 12 und 16).

Sei $\mathbf{P} \in \mathbb{K}^{s \times s}$ und $\mathbf{Q} \in \mathbb{K}^{s \times \tilde{m}}$ mit $\tilde{m} \leq m$ unabhängigen Spaltenvektoren. Die Spalten der

Algorithmus 14 Block-Arnoldi-Algorithmus

Eingabe: Matrizen \mathbf{P}, \mathbf{Q} ; Stufe des Block-Krylov-Unterraumes k ; Deflationstoleranz τ
Ausgabe: Basis \mathbf{V} des Block-Krylov-Unterraumes

```

1: function  $\mathbf{V} = \text{Block\_Arnoldi}(\mathbf{P}, \mathbf{Q}, k, \tau)$ 
2: Entfernung aller linear abhängigen Spaltenvektoren der Matrix  $\mathbf{Q}$  ▷ Initialisierungen
3:  $\tilde{m} = \text{Spaltenanzahl}(\mathbf{Q})$ 
4: Berechne die reduzierte  $\mathbf{QR}$ -Zerlegung:  $\mathbf{V}_0 \mathbf{H}_{1,0} = \mathbf{Q}$  ▷ Berechnung des ersten Blockes
5:  $\mathbf{V} = [\mathbf{V}_0]$ 
6: for  $j = 1 : k - 1$  ▷ Für  $k - 1$  weitere Blöcke
7:    $\mathbf{W} = \mathbf{P} \mathbf{V}_{j-1}$ 
8:   for  $i = 1 : j$  ▷ Orthogonalisierung
9:      $\mathbf{H}_{i,j} = \mathbf{V}_{i-1}^* \mathbf{W}$ 
10:     $\mathbf{W} = \mathbf{W} - \mathbf{V}_{i-1} \mathbf{H}_{i,j}$ 
11:  end for
12:  Berechne die Rank-Revealing- $\mathbf{QR}$ -Zerlegung:  $\mathbf{V}_j \mathbf{H}_{j+1,j} = \mathbf{W}$ ;  $t = \text{rang}(\mathbf{W}, \tau)$ 
13:  if  $t = 0$ 
14:    STOP ▷ Abbruch
15:  else
16:     $\mathbf{V}_j = \mathbf{V}_j(:, 1 : t)$ 
17:  end if
18:   $\mathbf{V} = [\mathbf{V} \ \mathbf{V}_j]$ 
19: end for
    
```

Matrix $\mathbf{V}_{[k]} = [\mathbf{V}_0 \ \mathbf{V}_1 \ \cdots \ \mathbf{V}_{k-1}]$ mit $\mathbf{V}_j \in \mathbb{K}^{s \times \tilde{m}}$, $j = 0, 1, \dots, k - 1$ bilden dann eine orthogonale Basis des Block-Krylov-Unterraumes (2.50), wenn alle oberen Dreiecksmatrizen $\mathbf{H}_{j+1,j}$, die im Ablauf des Algorithmus 14 berechnet werden, vollen Spaltenrang besitzen. Wie beim Arnoldi-Algorithmus in seiner Standardversion ist eine Reorthogonalisierung notwendig, damit die berechneten Spaltenvektoren der Matrix $\mathbf{V}_{[k]}$ orthogonal zueinander bleiben. Wird der Block-Arnoldi-Algorithmus mit einer Eingabematrix \mathbf{Q} , die m linear unabhängige Spaltenvektoren besitzt, verwendet, so erzeugt dieser eine Basis des Block-Krylov-Unterraumes $\mathcal{K}_k(\mathbf{P}, \mathbf{Q})$. Dieser kann als Vereinigung der m Krylov-Unterräume der Form (2.48) für jeden Spaltenvektor der Matrix \mathbf{Q} angesehen werden, siehe dazu auch Abschnitt 2.4. Die Berechnung der Basen der m Krylov-Unterräume erfolgt dabei auf dieselbe Weise, wie bei der Standardversion des Arnoldi-Algorithmus.

Für den Block-Arnoldi-Algorithmus besteht die Beziehung:

$$\mathbf{P} \mathbf{V}_{[k]} = \mathbf{V}_{[k]} \mathbf{H}_{[k]} + [\mathbf{0}, \dots, \mathbf{0}, \mathbf{V}_k \mathbf{H}_{k+1,k}], \quad (10.10)$$

wobei $\mathbf{H}_{[k]}$ eine obere Block-Hessenbergmatrix der Form

$$\mathbf{H}_{[k]} = \begin{bmatrix} \mathbf{H}_{1,1} & \mathbf{H}_{1,2} & \mathbf{H}_{1,3} & \cdots & \mathbf{H}_{1,k} \\ \mathbf{H}_{2,1} & \mathbf{H}_{2,2} & \mathbf{H}_{2,3} & \cdots & \mathbf{H}_{2,k} \\ 0 & \mathbf{H}_{3,2} & \mathbf{H}_{3,3} & \cdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \mathbf{H}_{k,k-1} & \mathbf{H}_{k,k} \end{bmatrix} \in \mathbb{K}^{(k \cdot \tilde{m}) \times (k \cdot \tilde{m})} \quad (10.11)$$

mit $\mathbf{H}_{i,j} \in \mathbb{K}^{\tilde{m} \times \tilde{m}}$ ist.

10.3.2 Ein modifizierter iterativer rationaler Krylov-Algorithmus (MIRKA)

Wie in Abschnitt 8.4.5 bereits dargestellt wurde, stellt bei den Krylov-Unterraum-Verfahren die optimale Wahl der Entwicklungsstellen und die Wahl der Stufe der dazu korrespondierenden Krylov-Unterräume ein offenes Problem dar. Die in Abschnitt 8.5 vorgestellten \mathcal{H}_2 -optimalen Verfahren IRKA und MIRIAM basieren auf einer iterativen Methode zum Auffinden optimaler Entwicklungsstellen. Die mathematisch fundierten Aussagen zur \mathcal{H}_2 -Optimalität können allerdings nur für Systeme erster Ordnung garantiert werden.

Zudem werden bei diesen Verfahren nur die Momente der Ordnung Null an den Entwicklungsstellen (entspricht der Übertragungsfunktion an den Entwicklungsstellen) berechnet, wodurch die Anzahl der in jeder Iteration des Verfahrens zu lösenden Gleichungssysteme gleich der Dimension des reduzierten Systems ist.

Da das Lösen dieser Gleichungssysteme die meiste Rechenzeit des gesamten Reduktionsprozesses in Anspruch nimmt, benötigen diese Verfahren deshalb bereits bei der Berechnung von Systemen mit moderater reduzierter Dimension viel Rechenzeit.

Das lässt die Anwendung dieser Verfahren auf Modelle, die für eine adäquate Approximation ein hinreichend großes reduziertes System ($r > 30$) benötigen, um die Übertragungsfunktion gut zu approximieren, als problematisch erscheinen.

Im folgenden Abschnitt wird eine neue Methode vorgestellt, die die Lage einer Anzahl von Anfangsentwicklungsstellen s_i , $i = 1, \dots, \hat{i}$ mit $\hat{i} < r$ iterativ optimiert. Im Unterschied zu den oben genannten Verfahren verwendet ein Reduktionsverfahren, das diese Methode zur Bestimmung der Entwicklungsstellen verwendet, neben den Momenten der nullten Ordnung noch Momente höherer Ordnung an den iterativ bestimmten Entwicklungsstellen.

Dadurch benötigt das Verfahren weniger Rechenzeit als die \mathcal{H}_2 -optimalen Verfahren. Zudem erfolgt die Bestimmung der benötigten Entwicklungsstellen, deren Anzahl \hat{i} durch die Anzahl der Anfangsentwicklungsstellen festgelegt wird, aus der Menge der $r \geq \hat{i}$ Eigenwerte des in der vorhergehenden Iteration reduzierten Systems, deren Imaginäranteile in einem Frequenzbereich mit hoher Priorität liegen, um so eine gute Approximation der Übertragungsfunktion in diesem Frequenzbereich zu erhalten.

In [85] wird die Wahl einer fest vorgegebenen Anzahl von Entwicklungsstellen s_i , $i = 1, \dots, r$ bei Krylov-Unterraummethoden anhand der Reduktion eines SISO-Systems erster Ordnung diskutiert.

Ausgehend von der Fehlerbestimmung bezüglich der \mathcal{H}_2 -Norm nach Satz 9.2 wird die folgende Wahl der Entwicklungsstellen empfohlen: Beginnend mit einem Satz Anfangsentwicklungsstellen s_i , $i = 1, \dots, r$ wird ein reduziertes System bestimmt. Darauf folgend wird ein neuer Satz Entwicklungsstellen für die Berechnung des reduzierten Systems der folgenden Iteration mittels $s_i = -\lambda_i$, $i = 1, \dots, r$ ermittelt, wobei $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_r|$ die geordneten Eigenwerte von $\hat{\mathbf{E}} - \lambda \hat{\mathbf{A}}$ und $\hat{\mathbf{E}}$, $\hat{\mathbf{A}}$ die Systemmatrizen des reduzierten Systems erster Ordnung sind.

Diese Methode zur Wahl neuer Entwicklungsstellen dient als Grundlage des Verfahrens IRKA (Algorithmus 10), das in Abschnitt 8.5 vorgestellt wurde und bei Konvergenz eine \mathcal{H}_2 -optimale Reduktion von Systemen erster Ordnung garantiert.

Das Verfahren MIRIAM (Algorithmus 11) stellt eine Erweiterung des Verfahrens IRKA für MIMO-Systeme dar, das auf tangentialer Interpolation basiert und bei Konvergenz eine \mathcal{H}_2 -optimale Reduktion von MIMO-Systemen erster Ordnung garantiert.

Aufbauend auf der Idee der iterativen Wahl neuer Entwicklungsstellen auf Basis der Eigenwerte bereits reduzierter Systeme wird hier ein modifiziertes Verfahren vorgeschlagen. Der hier wiedergegebenen Darstellung dieses Verfahrens dient ein Verfahren als Grundlage, das bereits in [57] und [58] veröffentlicht und *modifizierter iterativer rationaler Krylov-Algorithmus*⁵ (MIRKA) genannt wird.

In der ersten Iteration des Verfahrens MIRKA wird ausgehend von einem Satz Anfangswerte für \hat{i} Entwicklungsstellen s_i , $i = 1, \dots, \hat{i}$ ein reduziertes System bestimmt. In den darauf folgenden Iterationen des Verfahrens wird ein neuer Satz Entwicklungsstellen mittels $s_i = -\lambda_i$, $i = 1, \dots, \hat{i}$ ermittelt, wobei die \hat{i} neuen Entwicklungsstellen aus der geordneten Menge der Eigenwerte $|\operatorname{Im}(\lambda_1)| \leq |\operatorname{Im}(\lambda_2)| \leq \dots \leq |\operatorname{Im}(\lambda_r)|$ des reduzierten Systems ausgewählt werden.

Der Unterschied zwischen den Verfahren IRKA bzw. MIRIAM und dem modifizierten Verfahren MIRKA besteht demzufolge darin, dass für das modifizierte Verfahren anstatt alle r nur \hat{i} speziell ausgewählte Eigenwerte aus der Menge aller Eigenwerte des reduzierten Systems verwendet werden.

Dies verringert die Berechnungszeit des reduzierten Systems und erlaubt zudem eine Auswahl der für die Approximation eines bestimmten Frequenzintervalls besonders wichtigen Entwicklungsstellen zu treffen.

Das modifizierte Verfahren berechnet mit Hilfe jeder neu ausgewählten Entwicklungsstelle mehrere Spalten der Projektionsmatrizen \mathbf{V} und \mathbf{W} , die die Basis eines geeigneten Krylov-Unterraumes bilden.

Im Gegensatz zu IRKA bzw. MIRIAM wird so ein reduziertes System generiert, das nicht nur die Funktionswerte der Übertragungsfunktion (Momente 0-ter Ordnung), sondern auch Momente höherer Ordnung an den \hat{i} Entwicklungsstellen zur Übereinstimmung bringt.

Da das Lösen der Gleichungssysteme für zwei unterschiedliche Entwicklungsstellen neu erfolgen muss und dies die meiste Rechenzeit des gesamten Reduktionsprozesses in Anspruch nimmt, benötigt das modifizierte Verfahren deshalb bereits bei reduzierten Systemen mit moderater reduzierter Dimension bedeutend weniger Rechenzeit als die Verfahren IRKA und MIRIAM.

Zudem kann durch die Auswahl geeigneter neuer Entwicklungsstellen s_i , $i = 1, \dots, \hat{i}$ die Approximation in einem Frequenzintervall $[0, f_{\max}]$ optimiert werden.

Eine geeignete Auswahl der Entwicklungsstellen in Bezug auf die Reduktion von Strukturmodellen wird im folgenden dargestellt, wobei der, in den Veröffentlichungen des Autors [57] und [58] dargestellte, Auswahlprozess neuer Entwicklungsstellen in dieser Arbeit um eine Auswahl der, nach der Dominanz geordneten, Eigenwerte des reduzierten Systems erweitert wird.

⁵In Englisch: *Modified Iterative Rational Krylov Algorithm*

10.3.2.1 Auswahlprozess der Entwicklungsstellen

Die Auswahl geeigneter Entwicklungsstellen für eine folgende Iteration des hier vorgeschlagenen Verfahrens MIRKA erfolgt mit Hilfe der folgenden Schritte:

1. Die berechneten Eigenwerte des reduzierten Systems werden so geordnet, dass $|\operatorname{Im}(\lambda_1)| \leq |\operatorname{Im}(\lambda_2)| \leq \dots \leq |\operatorname{Im}(\lambda_r)|$ gilt, wobei mehrfach auftretende Eigenwerte nur einfach verwendet werden. Alle Eigenwerte, für die $|\operatorname{Im}(\lambda_j)| \leq 2\pi \cdot f_{\max}$ gilt, werden in den folgenden Schritten als potenzielle Entwicklungsstellen betrachtet.
2. Gilt für die Anzahl k der potenziellen Entwicklungsstellen $k < \hat{i}$, werden fehlende Eigenwerte aus der Menge der vorhandenen berechneten Eigenwerte hinzugenommen, für deren Imaginäranteil $|\operatorname{Im}(\lambda_j)| > 2\pi \cdot f_{\max}$ und $j > k$ gilt.
3. Um die Auswahl der Entwicklungsstellen nach der Dominanz zu erreichen, werden die Eigenwerte λ_j , $j = 2, \dots, k-1$ jeweils über das gewichtete Residuum

$$\mathbf{R}(\lambda_j) = \frac{(\mathbf{C}_p + \lambda_j \mathbf{C}_v) \mathbf{F}}{\lambda_j} \quad (10.12)$$

und mit Hilfe der Frobenius-Norm geordnet:

$$\|\mathbf{R}(\lambda_2)\|_F \geq \|\mathbf{R}(\lambda_3)\|_F \geq \dots \geq \|\mathbf{R}(\lambda_{k-1})\|_F. \quad (10.13)$$

4. Um eine Häufung von Entwicklungsstellen zu vermeiden und eine bessere Approximation in einem weiten Frequenzband zu erhalten, werden nur Eigenwerte als Entwicklungsstellen ausgewählt, deren Imaginäranteile einen Mindestabstand Δ zueinander aufweisen. Zur Erzielung einer guten Approximation an den Intervallgrenzen wird zunächst $s_1 = \lambda_1$ und $s_2 = \lambda_k$ gewählt. Die folgenden Entwicklungsstellen s_i , $i = 3, 4, \dots$ mit $|s_{i-1} - s_i| > \Delta$ werden daraufhin aus der Menge der Eigenwerte λ_j , $j = 2, \dots, k-1$ bestimmt. Werden dabei $k_s < \hat{i}$ neue Entwicklungsstellen gefunden, beginnt eine neue Auswahl der Entwicklungsstellen s_i , $i = 3, 4, \dots$ aus der Menge der Eigenwerte λ_j , $j = 2, \dots, k-1$ mit dem halbierten Wert des Mindestabstandes ($\Delta = \Delta/2$).

Zu Schritt 1: Bei den in dieser Arbeit entwickelten Verfahren werden die berechneten Eigenwerte mit $|\operatorname{Im}(\lambda_j)| \leq 2\pi \cdot f_{\max}$ als potenzielle Entwicklungsstellen verwendet, damit wird eine gute Approximation des unteren Frequenzbereichs erwartet, der in der praktischen Anwendung strukturmechanischer Modelle von besonderem Interesse ist.

Zu Schritt 3: Die Ordnung nach der Frobenius-Norm der gewichteten Residuen $\mathbf{R}(\lambda_j)$ der Eigenwerte λ_j , $j = 2, \dots, k-1$ dient dazu, die dominanten Eigenwerte des Systems zu betrachten.

Zu Schritt 4: Mit Δ wird auf die Verteilung der ausgewählten Entwicklungsstellen im ausgewählten Frequenzbereich Einfluß genommen. D.h. wenn gilt $|s_3 - s_2| < \Delta$ mit $s_3 = \operatorname{Im}(\lambda_2)$, erfolgt eine weitere Suche nach s_3 mit $s_3 = \operatorname{Im}(\lambda_3)$. Dieser Vorgang wird solange weitergeführt, bis eine Entwicklungsstelle mit $|s_3 - s_2| > \Delta$ gefunden wurde. Alle weiteren noch benötigten Entwicklungsstellen werden auf die gleiche Weise ermittelt. Für die Distanz wird in den Verfahren der Anfangswert $\Delta = (2\pi \cdot f_{\max})/(\hat{i} - 1)$ verwendet.

Sind durch den Auswahlprozess der vorhergehenden Schritte weniger neue Entwicklungsstellen als Anfangsentwicklungsstellen ermittelt worden ($k_s < \hat{i}$), wird der Wert für Δ sukzessive halbiert, um die Anzahl k_s der ausgewählten Entwicklungsstellen auf $k_s \geq \hat{i}$ zu erhöhen.

Wird durch dieses Vorgehen für Δ der Wert 1 unterschritten, erfolgt eine Wiederholung des Auswahlprozesses ab Schritt 2, wobei die Menge der nach Schritt 1 ermittelten k potenziellen Entwicklungsstellen um die Anzahl der fehlenden Entwicklungsstellen ($\hat{i} - k_s$) erhöht wird. Dies wird durch die Hinzunahme von vorhandenen Eigenwerten mit $|\text{Im}(\lambda_j)| > 2\pi \cdot f_{max}$ und $j > k$ aus der nach dem Imaginäranteil geordneten Menge der berechneten Eigenwerten erreicht.

Werden nach Durchlauf des gesamten Auswahlprozesses weniger neue Entwicklungsstellen als Anfangsentwicklungsstellen bestimmt ($k_s < \hat{i}$), so bricht der Auswahlprozess der Entwicklungsstellen mit der Warnung, dass nicht genügend Entwicklungsstellen ausgewählt werden konnten, den Reduktionsprozess ab.

10.3.2.2 Auswahl der Entwicklungsstellen zur Stabilisierung

Ein weiteres Problem bei der Reduktion von Systemen mit Krylov-Unterraum-Verfahren besteht darin, dass die Stabilität des reduzierten Systems nicht garantiert werden kann. In Veröffentlichungen wird in solchen Fällen vorgeschlagen, eine so genannte *implizit neugestartete*⁶ Krylov-Unterraum-Methode zu verwenden [82]. Allerdings sind diese Verfahren schwer zu implementieren und numerisch aufwendig. Als Alternative werden hier, wie es in [70, 81, 87] vorgeschlagen wird, Entwicklungsstellen gewählt, die weiter von der imaginären Achse entfernt liegen. Dies kann bei dem hier vorgeschlagenen Verfahren bereits bei der iterativen Auswahl der Entwicklungsstellen berücksichtigt werden. Führt die Reduktion mit einem Satz Entwicklungsstellen $s_1, \dots, s_{\hat{i}}$ mit $\text{Re}(s_i) < \text{Re}(s_{i+1})$, $i = 1, \dots, \hat{i} - 1$ zu einem instabilen reduzierten System, so wird eine untere Schranke $R = \text{Re}(s_1)$ für die Auswahl neuer Entwicklungsstellen definiert. Mit dieser Methode wird sichergestellt, dass ab der nächsten Iteration keine Eigenwerte λ_k mit $\text{Re}(\lambda_k) \leq R$ als neue Entwicklungsstellen verwendet werden. Eigenwerte mit kleinem Realteil werden somit mit zunehmender Anzahl an Iterationen des Verfahrens verworfen, bis nur noch Eigenwerte mit so großem Realteil verwendet werden, dass das System stabil reduziert wird. Wird die Schranke R für die Auswahl neuer Entwicklungsstellen im Ablauf der Entwicklungsstellenauswahl so gesetzt, dass ab der nächsten Iteration nicht mehr genug Eigenwerte λ_j mit $\text{Re}(\lambda_j) \geq R$ vorhanden sind, dann ist das System mit diesem Verfahren nicht stabil reduzierbar.

10.3.2.3 Auswahl der Entwicklungsstellen des Verfahrens MIRKA

Die beiden Auswahlmethoden lassen sich kombinieren und es ergibt sich der in Abbildung 10.5 als Ablaufdiagramm dargestellte und als Algorithmus 15 in Pseudocode wiedergegebene Algorithmus *Stellenauswahl*. Der dargestellte Algorithmus gibt alle geeigneten Eigenwerte aus, die als mögliche neue Entwicklungsstellen verwendet werden können. Der *modifizierte iterative rationale Krylov-Algorithmus* (MIRKA) verwendet dann aus den ausgewählten Eigenwerten die ersten \hat{i} Eigenwerte als neue Entwicklungsstellen.

⁶In Englisch: implizite restarted

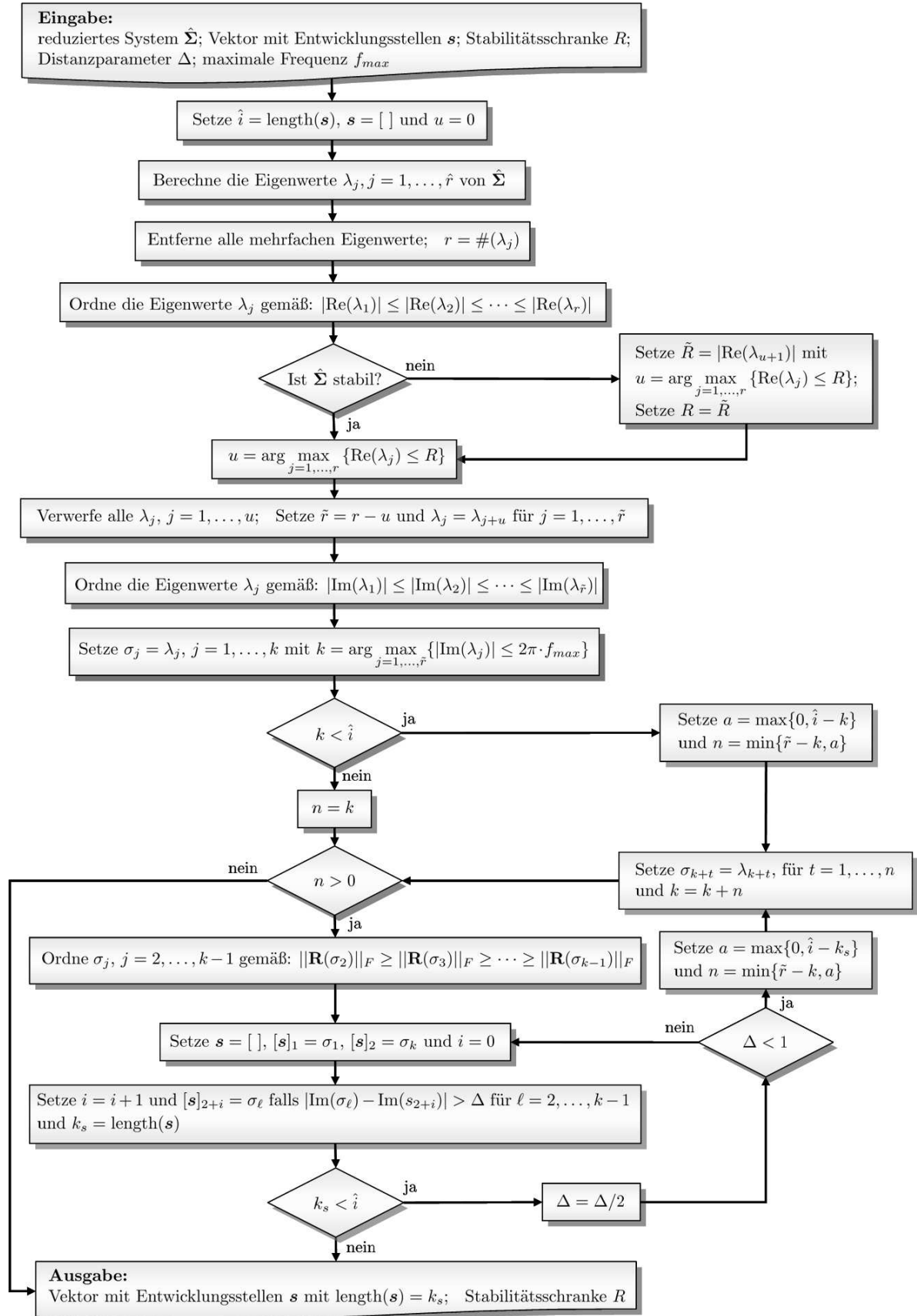


Abbildung 10.5: (Vereinfachtes) Ablaufdiagramm des Algorithmus *Stellenauswahl* zur Auswahl neuer Entwicklungsstellen

Algorithmus 15 Stellenauswahl

Eingabe: Reduziertes System $\hat{\Sigma}$ der Dimension \hat{n} ; Vektor mit Entwicklungsstellen \mathbf{s} ; Stabilitätsschranke R ;

Distanzparameter Δ ; maximale Frequenz f_{max}

Ausgabe: neue Entwicklungsstellen \mathbf{s} ; neue Stabilitätsschranke R_{neu}

```

1: function [ $\mathbf{s}, R$ ] = stellenauswahl( $\hat{\Sigma}, \mathbf{s}, R, \Delta, f_{max}$ )
2:  $u = 0$ ;  $i = 0$ ;  $\ell = 1$ ;  $\hat{i} = \text{length}(\mathbf{s})$ ;  $k_s = 0$ ;  $k = 0$ ;  $\mathbf{s} = []$  ▷ Initialisierungen
3: Berechne die Eigenwerte  $\lambda_j, j = 1, \dots, \hat{n}$  des reduzierten Systems  $\hat{\Sigma}$ 
4: Entferne alle mehrfachen Eigenwerte;  $r = \#(\lambda_j)$ 
5: Ordne die Eigenwerte in folgender Weise:  $|\text{Re}(\lambda_1)| \leq |\text{Re}(\lambda_2)| \leq \dots \leq |\text{Re}(\lambda_r)|$ 
6: if  $\hat{\Sigma}$  instabil ▷ Stabilisierung
7:   while  $\text{Re}(\lambda_\ell) \leq R, \ell = \ell + 1, \text{ end while}$ 
8:    $R = |\text{Re}(\lambda_\ell)|$  ▷ Aktualisierung von  $R$ 
9: end if
10: Verwerfe alle  $\lambda_j$  mit  $|\text{Re}(\lambda_j)| \leq R$ 
11:  $u = \arg \max_{j=1, \dots, r} \{\text{Re}(\lambda_j) \leq R\}$ ;  $r = r - u$ 
12: for  $j = 1 : r, \lambda_j = \lambda_{j+u}, \text{ end for}$ 
13: Ordne die Eigenwerte in folgender Weise:  $|\text{Im}(\lambda_1)| \leq |\text{Im}(\lambda_2)| \leq \dots \leq |\text{Im}(\lambda_r)|$ 
14: for  $j = 1 : r$  ▷ Auswahl nach der Frequenz
15:   if  $|\text{Im}(\lambda_j)| \leq 2\pi \cdot f_{max}$ 
16:      $k = k + 1$ ;  $\sigma_k = \lambda_j$ 
17:   end if
18: end for
19: if  $k < \hat{i}$  ▷ Wenn nicht genug Eigenwerte  $\lambda_j$  mit  $|\text{Im}(\lambda_j)| \leq 2\pi \cdot f_{max}$ 
20:    $a = \max\{0, \hat{i} - k\}$ ;  $n = \min\{r - k, a\}$  ▷ Bestimmung von  $n$  (Stellen mit  $|\text{Im}(\lambda_j)| > 2\pi \cdot f_{max}$ )
21:   for  $t = 1 : n$  ▷ Hinzunahme von  $n$  Stellen mit  $|\text{Im}(\lambda_j)| > 2\pi \cdot f_{max}$ 
22:      $k = k + 1$ ;  $\sigma_k = \lambda_k$ 
23:   end for
24: else
25:    $n = k$ 
26: end if
27: while  $n > 0$  ▷ Solange neue Eigenwerte als potenzielle Entwicklungsstellen
28:   Ordne die  $\sigma_j, j = 2, \dots, k-1$  gemäß:  $\|\mathbf{R}(\sigma_2)\|_F \geq \|\mathbf{R}(\sigma_3)\|_F \geq \dots \geq \|\mathbf{R}(\sigma_{k-1})\|_F$ 
29:   while  $k_s < \hat{i}$  ▷ Wenn zu wenig Entwicklungsstellen im relevanten Intervall
30:      $\mathbf{s} = []$ ;  $[\mathbf{s}]_1 = \sigma_1$ ;  $[\mathbf{s}]_2 = \sigma_k$ ;  $i = 0$ 
31:     for  $\ell = 2 : k-1$ 
32:       if  $|\text{Im}(\sigma_\ell) - \text{Im}([\mathbf{s}]_{2+i})| > \Delta$  ▷ Auswahl nach Distanz des Imaginäranteils
33:          $i = i + 1$ ;  $[\mathbf{s}]_{2+i} = \sigma_\ell$  ▷ Hinzunahme von neuen Entwicklungsstellen
34:       end if
35:     end for
36:      $k_s = \text{length}(\mathbf{s})$ 
37:     Reduziere  $\Delta$  ( $\Delta = \Delta/2$ )
38:     if  $\Delta < 1$  ▷ Wenn Distanz  $\Delta$  zu klein geworden ist
39:        $a = \max\{0, \hat{i} - k_s\}$ ;  $n = \min\{r - k, a\}$  ▷ Bestimmung von  $n$  (Stellen mit  $|\text{Im}(\lambda_j)| > 2\pi \cdot f_{max}$ )
40:       for  $t = 1 : n$  ▷ Hinzunahme von  $n$  Stellen mit  $|\text{Im}(\lambda_j)| > 2\pi \cdot f_{max}$ 
41:          $k = k + 1$ ;  $\sigma_k = \lambda_k$ 
42:       end for
43:       break ▷ Abbruch der while-Schleife: while  $k_s < \hat{i}$ 
44:     end if
45:   end while
46: end while
47: if  $k_s < \hat{i}$ , STOP, end if ▷ ABBRUCH: Nicht genug Entwicklungsstellen gefunden

```

10.3.2.4 Konvergenzkriterium

Als Abbruchkriterium der iterativen Entwicklungsstellenwahl wird die Differenz zweier aufeinander folgender reduzierter Systeme betrachtet. Dazu wird in der n -ten Iteration nach der Reduktion des Originalsystems die Differenz δ zum $(n-1)$ -ten reduzierten System in einer geeigneten Norm

$$\delta = \|\hat{\Sigma}_{n-1} - \hat{\Sigma}_n\| \quad (10.14)$$

betrachtet. Als geeignete Normen können die \mathcal{H}_∞ -Norm oder die \mathcal{H}_2 -Norm, wie sie in Abschnitt 3.5 vorgestellt wurden, verwendet werden. Für das Abbruchkriterium der implementierten Verfahren dieser Arbeit wird dazu die \mathcal{H}_2 -Norm verwendet. Soll ein vorgegebener Frequenzbereich $[f_{min}, f_{max}]$ approximiert werden, dann kommt als Alternative die Berechnung

$$\delta = \left(\int_{f_{min}}^{f_{max}} \|\mathbf{H}(2\pi i f) - \hat{\mathbf{H}}(2\pi i f)\|_F^2 df \right)^{1/2} \quad (10.15)$$

in Betracht. Die Bestimmung des Abbruchkriteriums mit dieser Methode ist dabei effizient durchführbar, weil dazu nur die Norm der Differenz zwischen zwei reduzierten Systemen berechnet werden muss.

Der Algorithmus 16 gibt den Algorithmus des Verfahrens MIRKA in Pseudocode wieder und die Abbildung 10.6 stellt den Ablauf des Algorithmus in der Form eines Ablaufdiagrammes dar. Zur Berechnung der Matrizen \mathbf{V} und \mathbf{W} in Zeile 6 des Algorithmus 16 können wahlweise Krylov-Unterraum-Verfahren, die auf dem Lanczos- oder dem Arnoldi-Algorithmus basieren und im Abschnitt 8.4.3 beschrieben wurden, verwendet werden.

Die Dimension des reduzierten Systems r ergibt sich in der Ausführung des Verfahrens als einseitiges Verfahren und wenn es zu keiner Elimination von Vektoren bei der Berechnung der Projektionsmatrizen \mathbf{V} und \mathbf{W} im Ablauf des Block-Arnoldi-Algorithmus kommt, durch die Anzahl der Anfangsentwicklungsstellen \hat{i} , der Stufe des Krylov-Unterraumes $k_i = [\mathbf{k}]_i$ an jeder Entwicklungsstelle $s_i = [\mathbf{s}]_i$, $i = 1, \dots, \hat{i}$, sowie der Anzahl der (linear unabhängigen) Eingänge \tilde{m} und Ausgänge \tilde{q} zu $r = \tilde{m} \cdot \sum_{i=1}^{\hat{i}} k_i$ bzw. zu $r = \tilde{q} \cdot \sum_{i=1}^{\hat{i}} k_i$. Wird das Verfahren als zweiseitiges Verfahren implementiert, ergibt sich für die Dimension des reduzierten Systems $r = \max\{\tilde{m}, \tilde{q}\} \cdot \sum_{i=1}^{\hat{i}} k_i$.

Die Anzahl der Momente der Übertragungsfunktion des reduzierten Systems, die mit den Momenten der Übertragungsfunktion des Originalsystems an jeder Entwicklungsstelle mindestens übereinstimmen, ergibt sich bei einem einseitigen Verfahren zu $r/(\tilde{m} \cdot \hat{i})$ bzw. zu $r/(\tilde{q} \cdot \hat{i})$ und bei einem zweiseitigen Verfahren zu $\lfloor r/(\tilde{m} \cdot \hat{i}) \rfloor + \lfloor r/(\tilde{q} \cdot \hat{i}) \rfloor$.

Da im Gegensatz zu den \mathcal{H}_2 -optimalen Verfahren die hier vorgeschlagene Methode zur Bestimmung neuer Entwicklungsstellen nicht alle iterativ bestimmten Eigenwerte verwendet, sind die theoretischen Bedingungen für eine \mathcal{H}_2 -Optimalität des Verfahrens nicht gegeben. Somit kann im Gegensatz zum Verfahren MIRIAM die \mathcal{H}_2 -Optimalität des Verfahrens nicht garantiert werden.

Mit diesem Verfahren ist es jedoch möglich, reduzierte Systeme zu berechnen, die im unteren Frequenzbereich bessere Approximationen an das Originalsystem darstellen als mit dem Verfahren MIRIAM. Zudem zeigte sich im Vergleich mit MIRIAM eine schnellere Konvergenz der

Eigenwerte des reduzierten Systems und damit einhergehend eine kürzere Berechnungszeit des reduzierten Systems.

Der Vorteil des hier beschriebenen Verfahrens, basierend auf der oben beschriebenen Auswahlmethode neuer Entwicklungsstellen, liegt in der Konvergenz der Entwicklungsstellen gegen Eigenfrequenzen des Systems, die im unteren Frequenzgang liegen, wodurch eine gute Approximation dieses Frequenzbereiches erzielt wird.

Diese Konvergenz der Entwicklungsstellen gegen die Eigenwerte im unteren Frequenzgang kann bei den \mathcal{H}_2 -optimalen Verfahren nicht garantiert werden, da die Eigenwerte, die als optimale Entwicklungsstellen identifiziert werden, in einem sehr hochdimensionalen Frequenzbereich liegen können. In diesem Fall erhält man eine gute Approximation des Frequenzganges in einem Frequenzbereich, der jedoch ausserhalb des Bereiches liegt, der für die hier betrachteten Anwendungen von Interesse ist.

Das Verfahren MIRKA kann für die Reduktion von Systemen erster Ordnung als *modifizierter iterativer rationaler Arnoldi-Algorithmus*⁷ (MIRA), siehe Algorithmus 17, oder in der Ausführung als *modifizierter iterativer rationaler Lanczos-Algorithmus*⁸ (MIRL), siehe Algorithmus 18, implementiert werden. Dazu kann in der Ausführung als MIRL der unsymmetrische (Block-) Lanczos-Algorithmus (Algorithmus 3) und in der Ausführung als MIRA der ein- oder (zwei-) seitige (Block-) Rational-Arnoldi-Algorithmus (Algorithmus 5) verwendet werden, um die Projektionsmatrizen \mathbf{V} und \mathbf{W} zu erzeugen.

Algorithmus 16 Modifizierter iterativer rationaler Krylov-Algorithmus (MIRKA)

Eingabe: System $\hat{\Sigma}_{1o}$ mit Systemmatrizen $\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}$; Vektor mit Anfangsentwicklungsstellen \mathbf{s} ; Vektor mit Stufen der Krylov-Unterräume \mathbf{k} ; Deflationstoleranz τ ; Toleranz $tol < 1$; Distanzparameter Δ ; maximale relevante Frequenz f_{max}

Ausgabe: Reduziertes System $\hat{\Sigma}_{1o}$ mit Systemmatrizen $\hat{\mathbf{E}}, \hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}$ der Dimension $r = \max\{\text{columns}(\mathbf{B}), \text{columns}(\mathbf{C})\} \cdot \sum_{i=1}^{\text{length}(\mathbf{s})} [\mathbf{k}]_i$

```

1: function  $[\hat{\mathbf{E}}, \hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}] = \text{MIRKA}(\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{s}, \mathbf{k}, \tau, tol, \Delta, f_{max})$ 
2:  $\delta = 1$ ;  $R = 0$ ;  $iter = 0$ ;  $\hat{i} = \text{length}(\mathbf{s})$ 
3: while  $\delta > tol$ 
4:    $iter = iter + 1$ 
5:    $\mathbf{V} = []$ ;  $\mathbf{W} = []$ 
6:   Berechne  $\mathbf{V}$  und  $\mathbf{W}$  mit einem Krylov-Unterraum-Verfahren
7:   Berechne das reduzierte System  $\hat{\Sigma}_{1o}$  mit der Kongruenztransformation (5.10)
8:    $[\mathbf{s}, R] = \text{stellenauswahl}(\hat{\Sigma}_{1o}, \mathbf{s}, R, \Delta, f_{max})$ ; Algorithmus 15 ▷ Neue Entwicklungsstellen
9:    $\mathbf{s} = \mathbf{s}(1 : \hat{i})$ 
10:   $\hat{\Sigma}_{1o,alt} \leftarrow \hat{\Sigma}_{1o}$ 
11:  if  $iter > 1$ 
12:    Berechne  $\delta = \|\hat{\Sigma}_{1o,alt} - \hat{\Sigma}_{1o}\|$  mit einer geeigneten Norm  $\|\cdot\|$ .
13:  end if
14: end while
```

⁷In Englisch: *Modified Iterative Rational Arnoldi Algorithm*

⁸In Englisch: *Modified Iterative Rational Lanczos Algorithm*

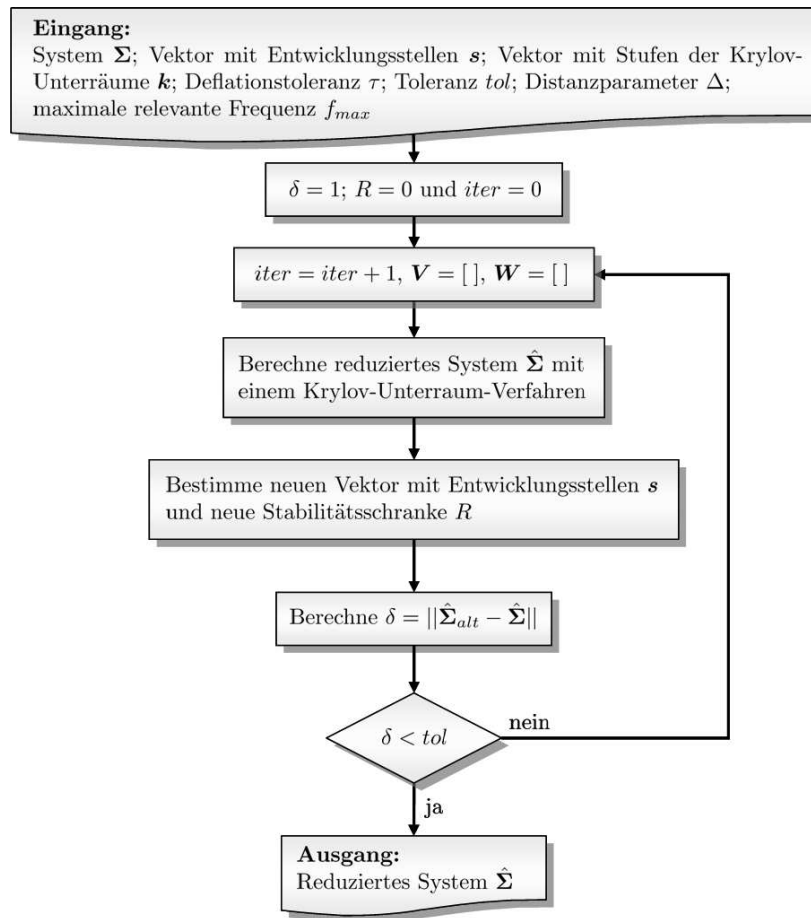


Abbildung 10.6: (Vereinfachtes) Ablaufdiagramm des *Modifizierten iterativen rationalen Krylov-Algorithmus* (MIRKA)

10.3.3 Anwendung auf Systeme zweiter Ordnung

Wie in den vorhergehenden Abschnitten schon dargestellt wurde, hat eine direkte Reduktion von Systemen zweiter Ordnung Vorteile bei der Verwendung des reduzierter Systeme zur Simulation von mechanischen Strukturen.

Die Verfahren IRKA und MIRIAM garantieren eine \mathcal{H}_2 -optimale Reduktion bislang nur für Systeme erster Ordnung. Im Gegensatz zu diesen beiden Verfahren lässt sich der im vorhergehenden Abschnitt vorgestellte Algorithmus durch leichte Anpassungen auch zur Reduktion von Systemen zweiter Ordnung anwenden. Dazu ist es nur nötig, in Algorithmus 16 Systemmatrizen eines Systems zweiter Ordnung zu verwenden, in Zeile 6 die Matrizen V und W mit Krylov-Unterraum-Verfahren für Systeme zweiter Ordnung durchzuführen und in Zeile 7 die Projektion mit (5.16) anstatt mit (5.10) durchzuführen.

Zur Erzeugung der Matrizen V und W können prinzipiell alle Krylov-Unterraum-Verfahren für Systeme zweiter Ordnung, wie sie in Abschnitt 8.4.4 dargestellt wurden, verwendet werden. Da in dieser Arbeit eine Anwendung der Algorithmen ausschließlich auf ungedämpfte oder proportional gedämpfte Systeme erfolgt, kann jedoch auf die Verwendung von Krylov-Unterraum-Verfahren, die auf Krylov-Unterräume zweiter Ordnung basieren, verzichtet werden. Die Berechnung der

Algorithmus 17 Modifizierter iterativer rationaler Arnoldi-Algorithmus (MIRA)

Eingabe: System Σ_{1o} mit Systemmatrizen E, A, B, C ; Vektor mit Anfangsentwicklungsstellen s ; Vektor mit Stufen der Krylov-Unterräume k ; Deflationstoleranz τ ; Toleranz $tol < 1$; Distanzparameter Δ ; maximale relevante Frequenz f_{max}

Ausgabe: Reduziertes System $\hat{\Sigma}_{1o}$ mit Systemmatrizen $\hat{E}, \hat{A}, \hat{B}, \hat{C}$ der Dimension $r = \max\{\text{columns}(B), \text{columns}(C)\} \cdot \sum_{i=1}^{\text{length}(s)} [k]_i$

```

1: function [ $\hat{E}, \hat{A}, \hat{B}, \hat{C}$ ] = MIRA( $E, A, B, C, s, k, \tau, tol, \Delta, f_{max}$ )
2:  $\delta = 1$ ;  $R = 0$ ;  $iter = 0$  ▷ Initialisierungen
3: while  $\delta > tol$  ▷ Solange keine Konvergenz
4:    $iter = iter + 1$ 
5:    $V = []$ ,  $W = []$ 
6:   [ $V, W$ ] = Rational_Arnoldi( $E, A, B, C, s, k, \tau$ ); Algorithmus 5
7:   Berechne das reduzierte System  $\hat{\Sigma}_{1o}$  mit der Projektion (5.10)
8:   [ $s, R$ ] = stellenauswahl( $\hat{\Sigma}_{1o}, s, R, \Delta, f_{max}$ ); Algorithmus 15 ▷ Neue Entwicklungsstellen
9:    $s = s(1 : \hat{i})$ 
10:   $\hat{\Sigma}_{1o,alt} \leftarrow \hat{\Sigma}_{1o}$ 
11:  if  $iter > 1$ 
12:    Berechne  $\delta = \|\hat{\Sigma}_{1o,alt} - \hat{\Sigma}_{1o}\|$  mit einer geeigneten Norm  $\|\cdot\|$ .
13:  end if
14: end while

```

Matrizen V und W orientiert sich deshalb an der Vorgehensweise, wie sie in [20] und [54] dargestellt wird. Dazu werden zur Berechnung der Matrizen V und W Algorithmen verwendet, die auf Krylov-Unterräumen erster Ordnung basieren. Im Falle eines ungedämpften Systems zweiter Ordnung werden dazu

$$P = -(s_i^2 M + K)^{-1} M \quad \text{und} \quad Q = (s_i^2 M + K)^{-1} F \quad (10.16)$$

in (2.50) zur Bestimmung von Basisvektoren des Krylov-Unterraumes

$$\begin{aligned} \mathcal{K}_k(-(s_i^2 M + K)^{-1} M, (s_i^2 M + K)^{-1} F) = \\ \mathcal{G}_k(-(s_i^2 M + K)^{-1} (2s_i M), -(s_i^2 M + K)^{-1} M, (s_i^2 M + K)^{-1} F) \end{aligned} \quad (10.17)$$

und im Falle eines proportional gedämpften Systems zweiter Ordnung werden dazu

$$P = -(s_i^2 M + s_i D + K)^{-1} M \quad \text{und} \quad Q = (s_i^2 M + s_i D + K)^{-1} F \quad (10.18)$$

in (2.50) zur Bestimmung von Basisvektoren des Krylov-Unterraumes

$$\mathcal{K}_k(-\tilde{K}^{-1} M, \tilde{K}^{-1} F) = \mathcal{G}_k(-\tilde{K}^{-1} (\alpha M + \beta K), -\tilde{K}^{-1} M, \tilde{K}^{-1} F) \quad (10.19)$$

verwendet, siehe dazu auch Abschnitt 8.4.4.6.

Algorithmus 19 gibt das zur Reduktion von ungedämpften oder proportional gedämpften Systemen zweiter Ordnung angepasste und implementierte Verfahren MIRA_2o in Pseudocode wieder.

Die Dimension des reduzierten Systems r ergibt sich, wenn es zu keiner Elimination linear abhängiger Spaltenvektoren während des Ablaufes des (Block_)Arnoldi-Algorithmus kommt, durch die Anzahl der Anfangsentwicklungsstellen \hat{i} , der Stufe des Krylov-Unterraumes $k_i = [k]_i$ an jeder Entwicklungsstelle $s_i = [s]_i$, $i = 1, \dots, \hat{i}$ und der Anzahl der (linear unabhängigen) Eingänge \tilde{m} zu $r = \tilde{m} \cdot \sum_{i=1}^{\hat{i}} k_i$.

Algorithmus 18 Modifizierter iterativer rationaler Lanczos-Algorithmus (MIRL)

Eingabe: System Σ_{1o} mit Systemmatrizen E, A, B, C ; Vektor mit Anfangsentwicklungsstellen s ; Vektor mit Stufen der Krylov-Unterräume k ; Deflationstoleranz τ ; Toleranz $tol < 1$; Distanzparameter Δ ; maximale relevante Frequenz f_{max}

Ausgabe: Reduziertes System $\hat{\Sigma}_{1o}$ mit Systemmatrizen $\hat{E}, \hat{A}, \hat{B}, \hat{C}$ der Dimension $r = \max\{\text{columns}(B), \text{columns}(C)\} \cdot \sum_{i=1}^{\text{length}(s)} [k]_i$

```

1: function  $[\hat{E}, \hat{A}, \hat{B}, \hat{C}] = \text{MIRL}(E, A, B, C, s, \hat{k}, \tau, tol, \Delta, f_{max})$ 
2:  $\delta = 1$ ;  $R = 0$ ;  $iter = 0$ ;  $\hat{i} = \text{length}(s)$  ▷ Initialisierungen
3: while  $\delta > tol$  ▷ Solange keine Konvergenz
4:    $iter = iter + 1$ 
5:    $V = [ ]$ ;  $W = [ ]$ 
6:   for  $i = 1 : \hat{i}$  ▷ Für jede Entwicklungsstelle
7:     Berechne die  $LU$ -Zerlegung:  $LU = (A - [s]_i E)$ 
8:      $P = U \setminus (L \setminus E)$ 
9:      $Q = U \setminus (L \setminus B)$ 
10:     $\tilde{P} = L^* \setminus (U^* \setminus E^*)$ 
11:     $\tilde{Q} = L^* \setminus (U^* \setminus C^*)$ 
12:     $[V_i, W_i] = \text{Lanczos}(P, \tilde{P}, Q, \tilde{Q}, [k]_i, \tau)$ ; Algorithmus 4
13:     $V = [V \ V_i]$  und  $W = [W \ W_i]$  ▷ Erweiterung der Projektionsmatrizen
14:     $W = (W^* V)^{-1} W^*$  ▷ Biorthogonalisierung
15:  end for
16:  Berechne das reduzierte System  $\hat{\Sigma}_{1o}$  mit der Projektion (5.10)
17:   $[s, R] = \text{stellenauswahl}(\hat{\Sigma}_{2o}, s, R, \Delta, f_{max})$ ; Algorithmus 15 ▷ Neue Entwicklungsstellen
18:   $s = s(1 : \hat{i})$ 
19:   $\hat{\Sigma}_{1o, alt} \leftarrow \hat{\Sigma}_{1o}$ 
20:  if  $iter > 1$ 
21:    Berechne  $\delta = \|\hat{\Sigma}_{1o, alt} - \hat{\Sigma}_{1o}\|$  mit einer geeigneten Norm  $\|\cdot\|$ 
22:  end if
23: end while

```

10.3.4 Diskussion der Approximationseigenschaften

Um die Approximationseigenschaften der reduzierten System, die mit Hilfe der in diesem Kapitel vorgestellten Verfahren generiert wurden, zu demonstrieren, wird das erste Beispielmmodell herangezogen, das in Abschnitt 10.1.2.1 beschrieben wurde. Dazu wird als Frequenzbereich das Intervall $[0 \text{ Hz}, 500 \text{ Hz}]$ betrachtet.

Um zu untersuchen, welche Vorteile bzw. Nachteile die Verfahren basierend auf der oben beschriebenen Auswahlmethode neuer Entwicklungsstellen gegenüber dem Verfahren MIRIAM aufweisen, wird das erste Testmodell, das vier Eingänge und sechs Ausgänge besitzt, mit dem Verfahren MIRA_2o reduziert und zum einen mit einem Modell, das mit Hilfe eines \mathcal{H}_2 -optimalen Verfahrens reduziert wurde und zum anderen mit einem modal reduzierten Modell der gleichen Dimension, verglichen. Da in dem Anwendungsgebiet, das in dieser Arbeit betrachtet wird, MIMO-Systeme reduziert werden müssen, wird als \mathcal{H}_2 -optimales Verfahren MIRIAM (Algorithmus 11) verwendet.

Der Algorithmus des Verfahrens MIRA_2o (Algorithmus 19) wird mit den $\hat{i} = 3$ Anfangsentwicklungsstellen $i \cdot 2\pi$, $250i \cdot 2\pi$ und $500i \cdot 2\pi$ sowie $\tau = 10^{-30}$, $tol = 10^{-5}$, $f_{max} = 500$ und $\Delta = (500/2) \cdot 2\pi$ verwendet. Zudem wird an jeder Entwicklungsstelle s_i , $i = 1, \dots, \hat{i}$ für die

Algorithmus 19 Modifizierter iterativer rationaler Arnoldi-Algorithmus für Systeme zweiter Ordnung (MIRA_2o)

Eingabe: System Σ_{2o} mit Systemmatrizen M, D, K, F, C_p, C_v ; Vektor mit Anfangsentwicklungsstellen s ; Vektor mit Stufen der Krylov-Unterräume k ; Deflationstoleranz τ ; Toleranz $tol < 1$; Distanzparameter Δ ; maximale relevante Frequenz f_{max}

Ausgabe: Reduziertes System $\hat{\Sigma}_{2o}$ mit Systemmatrizen $\hat{M}, \hat{D}, \hat{K}, \hat{F}, \hat{C}_p, \hat{C}_v$ der Dimension r

```

1: function [  $\hat{M}, \hat{D}, \hat{K}, \hat{F}, \hat{C}_p, \hat{C}_v$  ] = MIRA_2o(  $M, D, K, F, C_p, C_v, s, k, \tau, tol, \Delta, f_{max}$  )
2:  $\delta = 1$ ;  $R = 0$ ;  $iter = 0$ ;  $\hat{i} = \text{length}(s)$  ▷ Initialisierungen
3: Entfernung aller linear abhängigen Spaltenvektoren der Matrix  $F$ 
4: while  $\delta > tol$  ▷ Solange keine Konvergenz
5:    $iter = iter + 1$ 
6:    $V = [ ]$ 
7:   for  $i = 1 : \hat{i}$  ▷ Für jede Entwicklungsstelle
8:     Berechne die LU-Zerlegung:  $LU = ([s]_i^2 M + K)$  bzw.  $LU = ([s]_i^2 M + [s]_i D + K)$ 
9:      $P = -U \setminus (L \setminus M)$ 
10:     $Q = U \setminus (L \setminus F)$ 
11:     $V_i = (\text{Block\_})\text{Arnoldi}(P, Q, [k]_i, \tau)$ ; Algorithmus 3 (oder Algorithmus 14)
12:   end for
13:    $\tilde{V} = [V_1 \ V_2 \ \dots \ V_i]$  ▷ (Re-) Orthonormalisierung
14:   Berechne die Rank-Revealing-QR-Zerlegung:  $VR = \tilde{V}$ ,  $t = \text{rang}(\tilde{V}, \tau)$ 
15:    $V = V(:, 1 : t)$ 
16:    $W = V$ 
17:   Berechne das reduzierte System  $\hat{\Sigma}_{2o}$  mit der Projektion (5.16)
18:    $[s, R] = \text{stellenauswahl}(\hat{\Sigma}_{2o}, s, R, \Delta, f_{max})$ ; Algorithmus 15 ▷ Neue Entwicklungsstellen
19:    $s = s(1 : \hat{i})$ 
20:    $\hat{\Sigma}_{2o, alt} \leftarrow \hat{\Sigma}_{2o}$ 
21:   if  $iter > 1$ 
22:     Berechne  $\delta = \|\hat{\Sigma}_{2o, alt} - \hat{\Sigma}_{2o}\|$  mit einer geeigneten Norm  $\|\cdot\|$ .
23:   end if
24: end while

```

Stufe des Krylov-Unterraumes $[k]_i = \hat{k} = 2$ für $i = 1, \dots, \hat{i}$ gewählt. Dadurch erhält man ein reduziertes System zweiter Ordnung der Dimension $r = m \cdot \hat{i} \cdot \hat{k} = 4 \cdot 3 \cdot 2 = 24$. Die Momente der Übertragungsfunktion des reduzierten Systems mit dieser Dimension stimmen dann mit mindestens den ersten $r/(m \cdot \hat{i}) = 24/(4 \cdot 3) = 2$ Momenten der Übertragungsfunktion des Originalsystems an den Entwicklungsstellen überein.

Das reduzierte Modell, das aus der Anwendung des Verfahren MIRA_2o auf das ungedämpfte System des Modells und anschließender proportionaler Dämpfung mit den reduzierten Systemmatrizen resultiert, wird hier MIRA_ND genannt, das reduzierte Modell das aus der direkten Anwendung auf das proportional gedämpfte System des Modells resultiert, wird hingegen als MIRA_PD bezeichnet.

Da die \mathcal{H}_2 -optimalen Verfahren für Systeme erster Ordnung entwickelt wurden, erfolgt ein Vergleich des mit den verschiedenen Verfahren reduzierten Testmodells anhand des äquivalenten Systems erster Ordnung. Um die Krylov-Unterraum reduzierten Modelle zweiter Ordnung mit reduzierter Dimension 24 mit dem Verfahren MIRIAM vergleichen zu können, werden die mit den Verfahren MIRA_ND und MIRA_PD reduzierten Modelle mit Hilfe der Transformation (3.6) auf

ein System erster Ordnung der Dimension 48 transformiert. Die Krylov-Unterraum-Verfahren werden zudem einem modal reduzierten Modell gegenüber gestellt. Das modal reduzierte System wird dazu zunächst mit Hilfe der FEM-Berechnungssoftware MSC.Nastran[®] als reduziertes System zweiter Ordnung der Dimension 24 generiert, welches der Anzahl der Eigenfrequenzen im Intervall $[0 \text{ Hz}, 1023.4 \text{ Hz}]$ entspricht, und anschließend in ein System erster Ordnung (MODAL_1o) der Dimension 48 transformiert.

Um ein mit dem Verfahren MIRIAM reduziertes System erster Ordnung der selben Dimension zu erhalten, benötigt der Algorithmus des Verfahrens MIRIAM 48 Anfangsentwicklungsstellen sowie 48 rechte und linke Anfangsrichtungsvektoren. Diese werden durch die Anwendung einer vorhergehenden Iteration des Rational-Arnoldi-Verfahrens auf das zuvor auf ein System erster Ordnung transformierte Originalsystem bestimmt. Dabei wird das Originalsystem auf ein System erster Ordnung der Dimension 48 reduziert. Dieses reduzierte System erster Ordnung besitzt die benötigte Anzahl an Eigenwerten sowie linken und rechten Eigenvektoren, die daraufhin als Anfangsentwicklungsstellen bzw. als rechte und linke Anfangsrichtungsvektoren des Verfahrens MIRIAM verwendet werden.

Das Verfahren MIRIAM benötigt 9 Iterationen und verwendet 48 Entwicklungsstellen, deren Imaginäranteil innerhalb des Intervalls $[0.11, 18.68 \cdot 10^3]$ liegen. Die Verfahren MIRA_ND und MIRA_PD benötigten 2 Iterationen und verwenden in der zweiten Iteration $1.59 \cdot 10^{-3} + 0.11i$, $0.24 + 75.68i$ und $10.38 + 497.59i$ als Entwicklungsstellen.

Die benötigten Iterationen und Berechnungszeiten der verschiedenen Reduktionsverfahren und die Approximationsfehler der reduzierten Systeme sind in der Tabelle 10.1 zusammengefasst. Da die Berechnung des modal reduzierten Systems mit des FEM-Programms MSC.Nastran[®] erfolgt, die bezüglich der Berechnungszeit und des Speicherbedarfs optimale Algorithmen verwendet, wird hier auf den Vergleich der Berechnungszeit des modal reduzierten Systems mit den übrigen reduzierten Systemen verzichtet.

Die Abbildung 10.7(a) gibt den Verlauf des relativen Fehlers $\epsilon_{rel}(f, \mathcal{I})$ (10.3) und die Abbildung 10.7(b) gibt den Verlauf des relativen Fehlers $\epsilon_{rel}(f)$ (10.5) der verschiedenen reduzierten Modelle im Intervall $[0 \text{ Hz}, 750 \text{ Hz}]$ wieder. In Abbildung 10.8 sind die Verläufe dieser relativen Fehler der verschiedenen reduzierten Modelle im Intervall $[0 \text{ Hz}, 10\,000 \text{ Hz}]$ wiedergegeben.

Tabelle 10.1: Ergebnisse der verwendeten Verfahren

Verfahren	Moden	Zeit [s]	Anzahl Iterationen	Fehler $\epsilon_{max}(\mathcal{I})$ (10.4)	Fehler ϵ_{max} (10.6)	Fehler ϵ_{rel} (10.4)
MODAL_1o	24	–	–	71.81	$4.20 \cdot 10^{-3}$	$15 \cdot 10^{-3}$
MIRIAM	–	1742	9	$9.04 \cdot 10^4$	0.10	$33.5 \cdot 10^{-3}$
MIRA_ND	–	4.95	2	2.22	$0.11 \cdot 10^{-3}$	$8.7 \cdot 10^{-3}$
MIRA_PD	–	6.53	2	2.21	$0.11 \cdot 10^{-3}$	$8.8 \cdot 10^{-3}$

Aus den Ergebnissen ist ersichtlich, dass mit Krylov-Unterraum-Verfahren gute Approximationen des Frequenzganges im gewählten Frequenzintervall erzielt werden können.

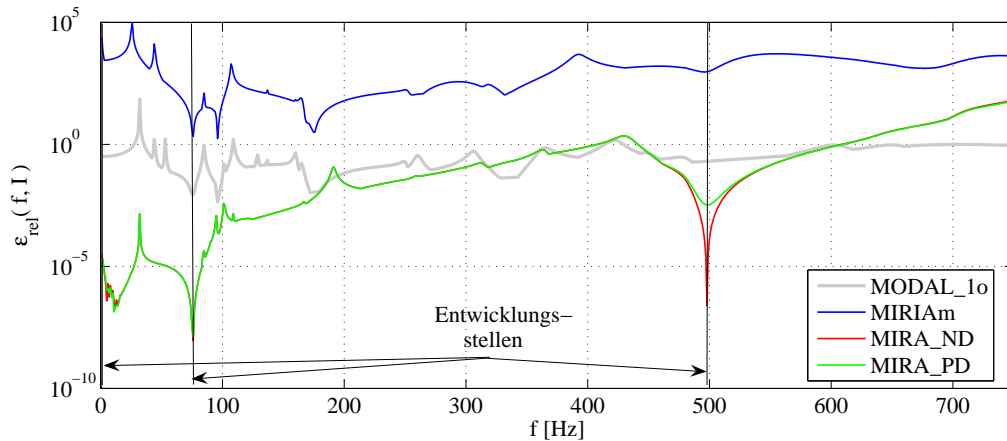
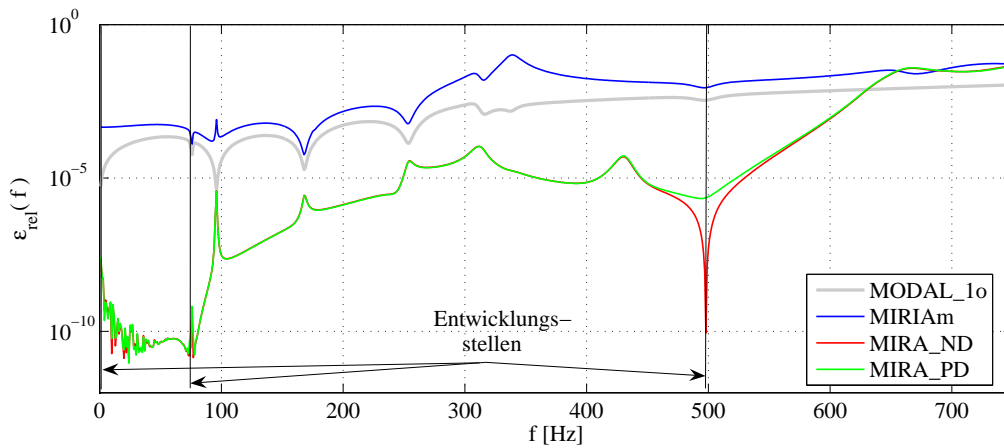

 (a) Relativer Approximationsfehler $\epsilon_{rel}(f, \mathcal{I})$

 (b) Relativer Approximationsfehler $\epsilon_{rel}(f)$

Abbildung 10.7: Relative Approximationsfehler der mit den Verfahren MODAL_2o, MIRIAM und MIRA_2o reduzierten Systeme - erstes Beispielmodell

Zudem zeigt der Vergleich, dass das Verfahren MIRA_2o bei Anwendung auf ungedämpfte (MIRA_ND) bzw. proportional gedämpfte Systeme (MIRA_PD) zu reduzierten Systemen führt, die bezüglich des Approximationsfehlers fast identisch sind. Die mit diesem Verfahren reduzierten Systeme weisen eine gute Approximation des Frequenzganges in der Umgebung der Entwicklungsstellen auf, wobei das reduzierte System MIRA_ND einen besonders geringen Fehler an den Entwicklungsstellen aufweist. Die Approximation kann allerdings für Frequenzen, die fern von den Entwicklungsstellen liegen, schnell schlechter werden.

Im Vergleich zum \mathcal{H}_2 -optimalen Verfahren MIRIAM führt die Anwendung des Verfahrens zu einer durchgehend besseren Approximation des gesamten gewählten Frequenzbereiches. Das Verfahren MIRIAM weist im gewählten Frequenzbereich bezüglich der hier verwendeten relativen Fehlermaße nicht mehr akzeptable Fehler auf.

In Abbildung 10.8(b) ist ersichtlich, dass das mit dem Verfahrens MIRIAM reduzierte System ab einer Frequenz, die außerhalb des relevanten Frequenzintervalls von 500 Hz liegt, eine bessere Approximation bezüglich des relativen Approximationsfehlers $\epsilon_{rel}(f)$ als die mit dem Verfahren

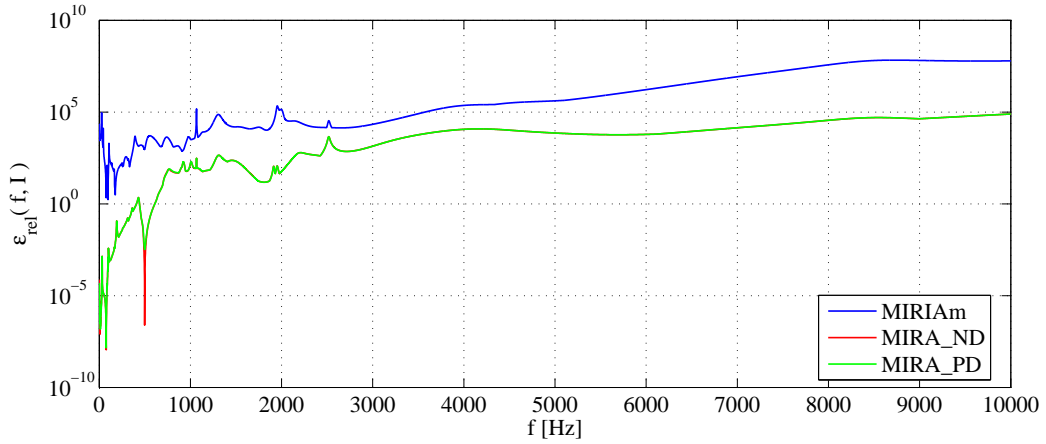
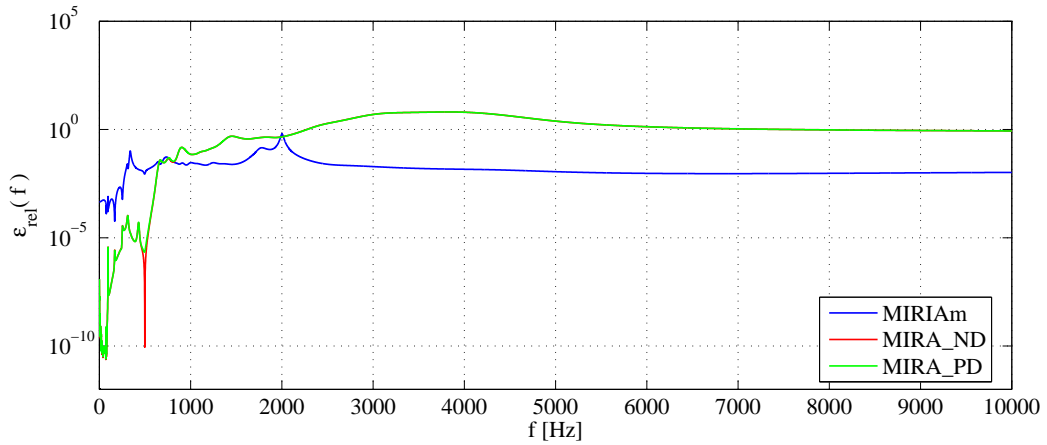
(a) Relativer Approximationsfehler $\epsilon_{rel}(f, \mathcal{I})$ (b) Relativer Approximationsfehler $\epsilon_{rel}(f)$

Abbildung 10.8: Relative Approximationsfehler der mit den Verfahren MIRIAm und MIRA_2o reduzierten Systeme bis 10000 Hz. - erstes Beispielmodell

MIRKA_2o reduzierten Systeme des Originalsystems darstellt. Bei Betrachtung der Abbildung 10.8(b) ist zu erkennen, dass das mit dem Verfahrens MIRIAm reduzierte System im gesamten Frequenzbereich bis 10000 Hz das Originalsystem schlechter approximiert als die mit dem Verfahren MIRKA_2o reduzierten Systeme.

Die Ursache für diese Beobachtungen liegt in der Konvergenz der Entwicklungsstellen gegen die Eigenwerte im unteren Frequenzbereich beim Verfahren MIRKA_2o. Bei dem \mathcal{H}_2 -optimalen Verfahren MIRIAm wird dies nicht erzielt, da die Eigenwerte, die als optimale Entwicklungsstellen identifiziert werden, im hochdimensionalen Frequenzbereich liegen.

Neben der besseren Approximation im gewählten Frequenzintervall benötigt das Verfahren mit der modifizierten Entwicklungsstellenwahl zudem bedeutend weniger Iterationen um zu konvergieren. Für die gewählte Dimension 48 werden beim Verfahren MIRIAm in jeder Iteration 48 Gleichungssysteme der Dimension $2n$ gelöst. Das Verfahren MIRA_2o benötigt hingegen nur 4 Berechnungen von Lösungen von Gleichungssystemen der Dimension n . In Verbindung mit einer höheren Anzahl an benötigten Iterationen, benötigt das Verfahren MIRIAm deshalb bedeutend

mehr Berechnungszeit als das Verfahren MIRA_2o.

Vergleicht man die Ergebnisse bezüglich des relativen Fehlers $\epsilon_{rel}(f, \mathcal{I})$, weisen die reduzierten Systeme MIRA_ND und MIRA_PD im Vergleich zum modal reduzierten Modell der gleichen Dimension im unteren Frequenzbereich deutlich bessere Approximationsergebnisse auf.

Vergleicht man die Ergebnisse bezüglich des relativen Fehlers $\epsilon_{rel}(f)$, weisen die mit den modifizierten Verfahren reduzierten Systeme (MIRA_ND, MIRA_PD) im betrachteten Frequenzbereich deutlich bessere Approximationsergebnisse als das modal reduzierte System auf.

Eine Ausnahme besteht bei der Frequenz 96 Hz, hier zeigen die Approximationen, die durch die Krylov-Unterraum-Verfahren reduzierten Modelle, ein wenig schlechtere Ergebnisse als das modal reduzierte System. Die Ursache dafür liegt darin begründet, dass an dieser Stelle eine Eigenmode des Systems liegt, die im Gegensatz zu den mit Krylov-Unterraum-Verfahren reduzierten Modellen im modal reduzierten Modell enthalten ist. Ein reduziertes Modell, das diese Stelle des Frequenzganges besser approximiert, wäre durch eine Erhöhung der Stufen der korrespondierenden Krylov-Unterräume zu den bestimmten Entwicklungsstellen in der Umgebung der Eigenfrequenz bei 96 Hz bestimmbar. Da bei den Verfahren MIRA_2o die Anzahl der Entwicklungsstellen und die Anzahl der Entwicklungen an jeder Entwicklungsstelle fest gewählt sind, erzielen diese Verfahren an besonders schwer zu approximierende Stellen des Frequenzganges schlechtere Ergebnisse.

10.3.5 Zusammenfassung der Resultate

Zusammenfassend kann festgestellt werden, dass mit MIRKA ein neues Reduktionsverfahren für Systeme erster und zweiter Ordnung entwickelt wurde, welches auf dem Lanczos- oder dem Arnoldi-Algorithmus zur Generierung der benötigten Projektionsmatrizen und auf einer speziellen Entwicklungsstellenwahl basiert. Die Entwicklungsstellenwahl ist dabei so ausgelegt, dass die Systeme, die im Anwendungsgebiet der Simulation von Werkzeugmaschinen auftreten, im unteren Frequenzbereich gut approximiert werden.

Ein Vergleich verschiedener Ausführungen dieses Verfahrens ist in den Veröffentlichungen [57] und [58] des Autors zu finden. Für das Anwendungsgebiet der Simulation von Werkzeugmaschinen bietet sich eine Reduktion des Originalsystems anhand eines Systems zweiter Ordnung an. In dieser Arbeit erfolgt deshalb eine Untersuchung des Verfahrens MIRKA in der Version als einseitiges Verfahren zur Reduktion von Systemen zweiter Ordnung, das auf dem Arnoldi-Algorithmus basiert. Dieses Verfahren wird hier MIRA_2o genannt und führt bei Anwendung auf die speziellen Systeme, die bei der Simulation von Werkzeugmaschinen auftreten, zu reduzierten Systemen, die das Originalsystem im relevanten unteren Frequenzbereich gut approximieren.

Die Anwendung von Verfahren mit der in dieser Arbeit betrachteten Methode der Entwicklungsstellenwahl führt zu reduzierten Systemen, die die Übertragungsfunktion des Originalsystems im relevanten Frequenzbereich besser approximieren als modal oder mit \mathcal{H}_2 -optimalen Verfahren reduzierten Systemen.

Die erzielten Resultate bestätigen den Vorteil des hier beschriebenen Verfahrens, basierend auf der Methode zur Auswahl neuer Entwicklungsstellen, der in der Konvergenz der Entwicklungs-

stellen gegen Eigenfrequenzen des Systems, die im unteren Frequenzgang liegen, besteht. Damit erzielt man mit diesen Verfahren eine gute Approximation des Frequenzganges in einem Frequenzbereich, der für die hier betrachteten Anwendungen von Interesse ist.

Diese Konvergenz der Entwicklungsstellen gegen die Eigenwerte im unteren Frequenzgang wird bei dem \mathcal{H}_2 -optimalen Verfahren MIRIAM nicht erzielt, da die Eigenwerte, die als optimale Entwicklungsstellen identifiziert werden, in einem sehr hochdimensionalen Frequenzbereich liegen. Man erhält eine gute Approximation des Frequenzganges in einem Frequenzbereich, der jedoch ausserhalb des Bereiches liegt, der für die hier betrachteten Anwendungen von Interesse ist, siehe dazu Abbildung 10.8(b). Durch die Verwendung einer geringeren Anzahl an Entwicklungsstellen als die \mathcal{H}_2 -optimalen Verfahren ist der Berechnungsaufwand des Verfahrens MIRA_2o zudem geringer als bei den \mathcal{H}_2 -optimalen Verfahren.

10.4 Verfahren basierend auf dem globalen Arnoldi-Algorithmus

Die im Anwendungsbereich der Simulation von Werkzeugmaschinen aus der FEM-Modellierung resultierenden Systeme besitzen in der Regel mehrere Ein- und Ausgänge. Da mit zunehmender Komplexität der modellierten Strukturen die Anzahl der Ein- bzw. Ausgänge zunimmt, sind Reduktionsverfahren, die MIMO-Systeme effizient reduzieren können, von Vorteil. Zur Lösung dieses Problems wird hier der sogenannte globale Arnoldi-Algorithmus herangezogen, der eine effiziente Version des Standard-Arnoldi-Algorithmus bei Anwendung auf MIMO-Systeme darstellt.

In diesem Abschnitt erfolgt deshalb die Herleitung eines Verfahrens, das auf einer Kombination der im vorhergehenden Kapitel beschriebenen Verfahren mit dem globalen Arnoldi-Algorithmus basiert.

Der globale Arnoldi-Algorithmus wurde als erstes in [90, 102] vorgestellt. Dabei waren die ersten Anwendungen dieses Algorithmus die Lösung linearer Gleichungen mit mehrfachen rechten Seiten und das Auffinden der Lösungen von Lyapunovgleichungen. Ein weiterer Anwendungsbereich ergab sich bei der Lösung von sehr großen Lyapunov-Matrixgleichungen [103]. Die Verwendung dieses Algorithmus im Kontext der Modellreduktion erfolgte in [39, 40, 41, 42] zur Reduktion von MIMO-Systemen erster Ordnung.

Auf diesen Arbeiten aufbauend wird in diesem Kapitel ein neues Verfahren zur Reduktion von MIMO-Systemen zweiter Ordnung, basierend auf dem globalen Arnoldi-Algorithmus, vorgestellt.

10.4.1 Der Globale Arnoldi-Algorithmus

10.4.1.1 Notationen

Zur weiteren Beschreibung des globalen Arnoldi-Algorithmus werden die folgenden Notationen und Definitionen benötigt. Das innere Produkt $\langle \Psi, \Upsilon \rangle_F$ zweier Vektoren (Matrizen) mit $\Psi, \Upsilon \in \mathbb{K}^{n \times s}$ ist durch $\langle \Psi, \Upsilon \rangle_F = \text{spur}(\Psi^* \Upsilon)$ gegeben. Die durch dieses innere Produkt induzierte Norm ist die Frobenius-Norm, die mit $\|\cdot\|_F$ bezeichnet wird.

Eine Menge bestehend aus Vektoren (Matrizen) aus $\mathbb{K}^{n \times s}$ wird als F -orthonormal bezeichnet, wenn die Vektoren (Matrizen) orthonormal bezüglich des inneren Produktes $\langle \cdot, \cdot \rangle_F$ sind.

Definition 10.1 (vec-Funktion) Für eine Matrix $\Upsilon \in \mathbb{K}^{n \times s}$ wird mit

$$\text{vec}(\Upsilon) = ([\Upsilon]_{*1}^* [\Upsilon]_{*2}^* \cdots [\Upsilon]_{*s}^*)^T \in \mathbb{K}^{ns} \quad (10.20)$$

die *vector stacking Operation* oder *vec-Funktion* definiert [95], wobei $\Upsilon = [[\Upsilon]_{*1} [\Upsilon]_{*2} \cdots [\Upsilon]_{*s}] \in \mathbb{K}^{n \times s}$ und $[\Upsilon]_{*j} \in \mathbb{K}^n$ für $j = 1, \dots, s$.

Definition 10.2 (Kronecker-Produkt) Für zwei Matrizen $\Upsilon \in \mathbb{K}^{s \times s}$ und $\Psi \in \mathbb{K}^{t \times t}$ wird mit \otimes das Kronecker-Produkt

$$\Upsilon \otimes \Psi = \begin{pmatrix} [\Upsilon]_{11} \Psi & [\Upsilon]_{12} \Psi & \cdots & [\Upsilon]_{1s} \Psi \\ [\Upsilon]_{21} \Psi & [\Upsilon]_{22} \Psi & \cdots & [\Upsilon]_{2s} \Psi \\ \vdots & \vdots & \ddots & \vdots \\ [\Upsilon]_{s1} \Psi & [\Upsilon]_{s2} \Psi & \cdots & [\Upsilon]_{ss} \Psi \end{pmatrix} = [[\Upsilon]_{ij} \Psi]_{i,j=1}^s \in \mathbb{K}^{st \times st} \quad (10.21)$$

bezeichnet.

Anmerkung 10.1 Mit den obigen Definitionen lassen sich folgende Beziehungen ableiten:

- a) $\text{vec}(\Upsilon \Psi \Omega) = (\Omega^* \otimes \Upsilon) \text{vec}(\Psi)$
- b) $\text{vec}(\Upsilon)^* \text{vec}(\Psi) = \text{spur}(\Upsilon^* \Psi)$
- c) $(\Upsilon \otimes \Psi)^* = \Upsilon^* \otimes \Psi^*$
- d) $(\Upsilon \otimes \Psi)^j = \Upsilon^j \otimes \Psi^j$

BEWEIS: Siehe z. B. [77].

10.4.1.2 Der globale Arnoldi-Algorithmus

Der globale Arnoldi-Algorithmus ist eine Modifikation des klassischen (Block-) Arnoldi-Algorithmus. Der hauptsächliche Unterschied zwischen beiden Varianten liegt darin, dass beim globalen Arnoldi-Algorithmus das innere Produkt zweier Vektoren (Matrizen) $\langle \mathbf{P}, \mathbf{Q} \rangle = \mathbf{P}^* \mathbf{Q}$ mit $\mathbf{P}, \mathbf{Q} \in \mathbb{K}^{n \times s}$ durch $\langle \mathbf{P}, \mathbf{Q} \rangle_F = \text{spur}(\mathbf{P}^* \mathbf{Q})$ ersetzt ist und die dadurch induzierte Frobenius-Norm verwendet wird. Wenn $s = 1$ gilt, sind der globale Arnoldi-Algorithmus und der klassische Arnoldi-Algorithmus identisch.

Der globale Arnoldi-Algorithmus [102], der hier leicht modifiziert als Algorithmus 20 dargestellt ist, generiert eine F -orthonormale Basis $\mathbf{V} = [\mathbf{V}_0 \mathbf{V}_1 \dots \mathbf{V}_{k-1}]$, wobei $\mathbf{V}_j \in \mathbb{K}^{n \times s}$, des Krylov-Unterraumes $\mathcal{K}_k(\mathbf{P}, \mathbf{Q})$ mit $\mathbf{P} \in \mathbb{K}^{n \times n}$ und $\mathbf{Q} \in \mathbb{K}^{n \times s}$, d.h. es gilt:

$$\begin{aligned} \langle \mathbf{V}_i, \mathbf{V}_j \rangle_F &= 0, \quad \text{für } i, j = 0, 1, \dots, k-1 \text{ mit } i \neq j \\ \langle \mathbf{V}_i, \mathbf{V}_j \rangle_F &= 1, \quad \text{für } i = j. \end{aligned} \quad (10.22)$$

Wie bereits bei der Blockversion des Arnoldi-Algorithmus dargestellt wurde, kann die Berechnung der Matrix \mathbf{Z} in Zeile 6 des Algorithmus zu einer Matrix mit linear abhängigen Spaltenvektoren führen. Um dies zu vermeiden wird der hier dargestellte globale Arnoldi-Algorithmus um eine Rank-Revealing- \mathbf{QR} -Zerlegung und einer Elimination linear abhängiger Spaltenvektoren ergänzt (Zeilen 7 und 8). Ist die Frobenius-Norm der Matrix \mathbf{W} kleiner als die Maschinengenauigkeit ϵ_M , dann bricht der Algorithmus in Zeile 15 ab.

Algorithmus 20 Globaler Arnoldi-Algorithmus nach [102]

Eingabe: Matrizen \mathbf{P}, \mathbf{Q} ; Stufe des Block-Krylov-Unterraumes k ; Deflationstoleranz τ

Ausgabe: Basis \mathbf{V} des (globalen) Krylov-Unterraumes

```

1: function  $[\mathbf{V}] = \text{Globaler\_Arnoldi}(\mathbf{P}, \mathbf{Q}, k, \tau)$ 
2: Entfernung aller linear abhängigen Spaltenvektoren der Matrix  $\mathbf{Q}$  ▷ Initialisierungen
3:  $h_{1,0} = \|\mathbf{Q}\|_F$ ;  $\mathbf{V}_0 = \mathbf{Q} / \|\mathbf{Q}\|_F$  ▷ Normierung
4:  $\mathbf{V} = [\mathbf{V}_0]$ 
5: for  $j = 1 : k - 1$ 
6:    $\mathbf{Z} = \mathbf{P}\mathbf{V}_{j-1}$ 
7:   Berechne die Rank-Revealing- $\mathbf{QR}$ -Zerlegung:  $\mathbf{W}\mathbf{R} = \mathbf{Z}$ ;  $t = \text{rang}(\mathbf{Z}, \tau)$ 
8:    $\mathbf{W} = \mathbf{W}(:, 1 : t)$ 
9:   for  $i = 1 : j$  ▷  $F$ -Orthogonalisierung
10:     $h_{i,j} = \text{spur}((\mathbf{V}_{i-1}(:, 1 : t))^* \mathbf{W})$ 
11:     $\mathbf{W} = \mathbf{W} - h_{i,j} \mathbf{V}_{i-1}(:, 1 : t)$ 
12:   end for
13:    $h_{j+1,j} = \|\mathbf{W}\|_F$ 
14:   if  $h_{j+1,j} \leq \epsilon_M$ 
15:     STOP ▷ Abbruch
16:   end if
17:    $\mathbf{V}_j = \mathbf{W} / h_{j+1,j}$  ▷ Normierung von  $\mathbf{W}$ 
18:    $\mathbf{V} = [\mathbf{V} \ \mathbf{V}_j]$  ▷ Erweiterung der Matrix  $\mathbf{V}$ 
19: end for
```

10.4.1.3 Eigenschaften des globalen Arnoldi-Algorithmus

Sei $\mathbf{V}_{(k)} = [\mathbf{V}_0 \ \mathbf{V}_1 \ \dots \ \mathbf{V}_{k-1}] \in \mathbb{K}^{n \times r}$, $r = k \cdot s$ und $\mathbf{H}_{(k)}$ die korrespondierende $k \times k$ obere Hessenberg-Matrix. Dann gilt die folgende Beziehung

$$\mathbf{P}\mathbf{V}_{(k)} = \mathbf{V}_{(k)}(\mathbf{H}_{(k)} \otimes \mathbf{I}_s) + h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T, \quad (10.23)$$

mit

$$\mathbf{H}_{(k)} = \begin{bmatrix} h_{1,1} & h_{1,2} & h_{1,3} & \cdots & h_{1,k} \\ h_{2,1} & h_{2,2} & h_{2,3} & \cdots & h_{2,k} \\ 0 & h_{3,2} & h_{3,3} & \cdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_{k,k-1} & h_{k,k} \end{bmatrix} \in \mathbb{K}^{k \times k} \quad (10.24)$$

und $\mathbf{E}_k = \mathbf{e}_k \otimes \mathbf{I}_s$ wobei \mathbf{e}_k den (k) -ten Einheitsvektor mit geeigneter Dimension bezeichnet.

Man beachte, dass die Hessenberg-Matrix $\mathbf{H}_{(k)}$, die durch den globalen Arnoldi-Algorithmus generiert wird, die Dimension $k \times k$ hat. Im Gegensatz dazu ist die Matrix $\mathbf{H}_{[k]}$ in Gleichung

(10.10), die durch den Block-Arnoldi-Algorithmus erzeugt wird, eine Block-Hessenberg-Matrix der Dimension $k \cdot \tilde{m} \times k \cdot \tilde{m}$. Darüber hinaus haben lineare Abhängigkeiten zwischen den Spaltenvektoren der generierten Matrizen \mathbf{V}_i , $i = 0, 1, \dots, k-1$ keine Auswirkungen auf den globalen Arnoldi-Algorithmus, siehe [102].

Der größte Unterschied zwischen dem globalen und dem Block-Arnoldi-Algorithmus liegt in der berechneten Basis des Krylov-Unterraumes. Der globale Arnoldi-Algorithmus generiert eine F -orthonormale Basis des Matrix-Krylov-Unterraumes $\mathcal{K}_k(\mathbf{P}, \mathbf{Q}) \subset \mathbb{K}^{n \times s}$, während der Block-Arnoldi-Algorithmus eine orthonormale Basis des Block-Krylov-Unterraumes $\mathcal{K}_k(\mathbf{P}, \mathbf{Q}) \subset \mathbb{K}^n$ generiert. Die Matrizen, die mit dem Block-Arnoldi-Algorithmus berechnet werden, besitzen orthogonale Spaltenvektoren, während die Matrizen, die mit dem globalen Arnoldi-Algorithmus berechnet werden, eine F -orthonormale Basis besitzen.

Anmerkung 10.2 *Der globale Arnoldi-Algorithmus bricht in der j -ten Iteration in Zeile 15 genau dann ab, wenn $h_{j+1,j} = 0$ (oder kleiner als die Maschinengenauigkeit ϵ_M) in Zeile 13 des Algorithmus 20 gilt. In diesem Fall wird ein invarianter Unterraum erzeugt. Man spricht in diesem Fall von einem lucky breakdown. Bei der Block-Version des Arnoldi-Algorithmus können sogenannte serious breakdowns auftreten. In diesem Fall bricht der Algorithmus ab, ohne dass ein invarianter Unterraum erzeugt wurde. Dadurch sind bei der Implementierung der Block-Version des Arnoldi-Algorithmus Deflationstechniken notwendig, siehe [102].*

Im Vergleich zur Block-Version des Arnoldi-Algorithmus ist der globale Arnold-Algorithmus effizienter. Vergleicht man beide Algorithmen benötigt der Block-Arnoldi-Algorithmus in jeder Iteration eine \mathbf{QR} -Zerlegung der Matrix \mathbf{Q} bzw. \mathbf{W} (Zeilen 4 und 14), während der globale Arnoldi-Algorithmus in Zeile 17 nur die Division durch die Frobenius-Norm der Matrix \mathbf{W} benötigt. Darüber hinaus benötigt die Block-Version die Berechnung von $\mathbf{V}_i^* \mathbf{W}$ in Zeile 11, während der globale Arnoldi-Algorithmus in Zeile 10 nur die Berechnung von $\text{spur}(\mathbf{V}^T \mathbf{W})$ benötigt. Schließlich benötigt die Block-Version in Zeile 12 ein Matrix-Matrix-Produkt, während die globale Variante in Zeile 11 nur ein Skalar-Matrix-Produkt benötigt.

In der folgenden Tabelle 10.2 ist der numerische Aufwand des Block-Arnoldi und des globalen Arnoldi-Algorithmus vergleichend dargestellt. Dazu werden die Anzahl der benötigten Rechenoperationen beider Verfahren, die zur Berechnung von k Iterationen für den numerischen Aufwand hauptsächlich verantwortlich sind, gegenübergestellt. Die Berechnung des Block-Arnoldi-Algorithmus wird aufwendig, wenn die Anzahl k der Iterationen ansteigt, da der Block-Arnoldi-Algorithmus zur Orthogonalisierung der Spaltenvektoren den modifizierten Gram-Schmidt-Algorithmus mit Matrizen der Dimension $n \times m$ durchführt. Der globale Arnoldi-Algorithmus ist deshalb im Vorteil gegenüber der Blockversion des Arnoldi-Algorithmus, wenn die Anzahl der Iterationen k hoch ist und gleichzeitig der numerische Aufwand zur Berechnung der Matrix-Vektor-Operationen mit der Matrix \mathbf{Q} hoch ist [103].

Werden der globale Arnoldi-Algorithmus und der Block-Arnoldi-Algorithmus auf das selbe Matrixpaar (\mathbf{P}, \mathbf{Q}) angewendet, spannen die resultierenden Matrizen $\mathbf{V}_{(k)}$ und $\mathbf{V}_{[k]}$ beide den selben Krylov-Unterraum $\mathcal{K}_k(\mathbf{P}, \mathbf{Q})$ auf. Die Art der Orthonormierung der Basisvektoren von $\mathcal{K}_k(\mathbf{P}, \mathbf{Q})$ ist der einzige Unterschied in dem sich die durch den Block-Arnoldi-Algorithmus und

Tabelle 10.2: Aufwand des Block-Arnoldi bzw. des globalen Arnoldi-Algorithmus zur Berechnung von k Iterationen

Kosten (Operationen)	Block-Arnoldi	globaler Arnoldi
Matrix-Matrix-Multiplikation, $n \times n$ mit $n \times m$ Matrizen	$2 \cdot k \cdot n^2 \cdot m$ [Zeile 7]	$2 \cdot k \cdot n^2 \cdot m$ [Zeile 6]
Matrix-Matrix-Multiplikation, bzw. die Spur von $m \times n$ und $n \times m$ Matrizen	$\frac{k(k+1)}{2} \cdot 2 \cdot n \cdot m^2$ [Zeile 11]	$\frac{k(k+1)}{2} (2 \cdot n \cdot m + m)$ [Zeile 10]
Matrix-Matrix-Multiplikation $n \times m$ mit $m \times m$ Matrizen, bzw. Matrix-Multiplikation, $n \times m$ Matrix	$\frac{k(k+1)}{2} \cdot 2 \cdot n \cdot m^2$ [Zeile 12]	$\frac{k(k+1)}{2} (n \cdot m)$ [Zeile 11]
QR -Zerlegung einer $n \times m$ Matrix, bzw. Frobenius-Norm einer $n \times m$ Matrix	$3(k+1) \cdot m^3 \cdot (n - \frac{m}{3})$ [Zeilen 4 und 14]	$(k+1) \cdot (2 \cdot n \cdot m + m)$ [Zeilen 3 und 13]

dem globalen Arnoldi-Algorithmus berechnete Basis unterscheidet.

In [81, Kapitel 3] und darauf aufbauend in Kapitel 8 der vorliegenden Arbeit wird gezeigt, dass die Eigenschaft des Momentenabgleiches (8.33) nur von der Tatsache abhängt, ob die Spalten von $\mathbf{V} = \mathbf{V}_{[k]}$ bzw. $\mathbf{V} = \mathbf{V}_{(k)}$ einen geeigneten Krylov-Unterraum $\mathcal{K}_k(\mathbf{P}, \mathbf{Q})$ mit geeigneten Matrizenpaaren (\mathbf{P}, \mathbf{Q}) aufspannen. Dabei ist die Art der Berechnung von \mathbf{V} oder ob dessen Spaltenvektoren weitere Bedingungen erfüllen unerheblich. Der globale Arnoldi-Algorithmus lässt sich somit in gleicher Weise wie der Block-Arnoldi-Algorithmus zur numerisch stabilen Berechnung geeigneter Projektionsmatrizen \mathbf{V} und \mathbf{W} verwenden.

10.4.2 Der iterative rationale globale Arnoldi-Algorithmus (IRGA)

Da die Herleitung eines Reduktionsverfahrens, das auf dem globalen Arnoldi-Algorithmus basiert, anhand von Systemen erster Ordnung anschaulicher herzuleiten ist, erfolgt in diesem Abschnitt zunächst die Herleitung eines Reduktionsverfahrens für Systeme erster Ordnung. Die Herleitung orientiert sich dabei an die Darstellung in [39]. Anschließend erfolgt die Herleitung eines neuen Verfahrens, mit dem sich Systeme zweiter Ordnung reduzieren lassen und das auf dem globalen Arnoldi-Algorithmus basiert.

10.4.2.1 Der globale Arnoldi-Algorithmus für Systeme erster Ordnung

Für $\text{vec}(\mathbf{P}^j, \mathbf{Q})$ erhält man mit Anmerkung 10.1, a) das

$$\text{vec}(\mathbf{P}^j, \mathbf{Q}) = (\mathbf{I} \otimes \mathbf{P}^j) \text{vec}(\mathbf{Q}) \quad (10.25)$$

gilt. Mit einer geeigneten Wahl der Matrizen \mathbf{P} und \mathbf{Q} , wie sie in Abschnitt 8.4.3 beschrieben wurde, können die (Eingangs- bzw. Ausgangs-) Momente eines Systems erster Ordnung (3.56) somit als vec -Funktion aufgefasst werden.

Der globale Arnoldi-Algorithmus kann damit als klassischer Arnoldi-Algorithmus angewandt auf das Paar $((\mathbf{I} \otimes \mathbf{P}), \text{vec}(\mathbf{Q})) = (\mathbf{\Gamma}, \boldsymbol{\theta})$ mit $\mathbf{\Gamma} \in \mathbb{K}^{2nm \times 2nm}$ und $\boldsymbol{\theta} \in \mathbb{K}^{2nm}$ angesehen werden, der eine Basis $\mathbf{V} \in \mathbb{K}^{2nm \times k}$ des Krylov-Unterraumes

$$\mathcal{K}_k((\mathbf{I} \otimes \mathbf{P}), \text{vec}(\mathbf{Q})) = \text{colspan}\{\text{vec}(\mathbf{Q}), (\mathbf{I} \otimes \mathbf{P})\text{vec}(\mathbf{Q}), \dots, (\mathbf{I} \otimes \mathbf{P})^{k-1}\text{vec}(\mathbf{Q})\} \quad (10.26)$$

berechnet und dabei das innere Produkt $\langle \mathbf{P}, \mathbf{Q} \rangle_F = \text{vec}(\mathbf{P})^T \text{vec}(\mathbf{Q})$ verwendet.

Anstatt diese Berechnungen mit Hilfe der vec -Funktion tatsächlich durchzuführen, werden, wie bei der Blockversion des Arnoldi-Algorithmus, die Spaltenvektoren blockweise berechnet. Durch Ersetzen des inneren Produktes $\langle \mathbf{P}, \mathbf{Q} \rangle_F$ nach Anmerkung 10.1, b) durch

$$\langle \mathbf{P}, \mathbf{Q} \rangle_F = \text{vec}(\mathbf{P})^* \text{vec}(\mathbf{Q}) = \text{spur}(\mathbf{P}^* \mathbf{Q}) \quad (10.27)$$

erzeugt der Algorithmus eine Basis $\mathbf{V} \in \mathbb{K}^{2n \times k \cdot m}$, die einen Krylov-Unterraum aufspannt, der mit dem Krylov-Unterraum identisch ist, der durch die Basis aufgespannt wird, die durch den Block-Arnoldi-Algorithmus erzeugt wird. Somit gilt die Eigenschaft des Momentenabgleiches sowohl für Reduktionsmethoden, die auf dem globalen Arnoldi-Algorithmus basieren ebenso wie für die Reduktionsmethoden, die auf dem Block-Arnoldi-Algorithmus basieren. Anhand dieser Eigenschaft kann ein Reduktionsverfahren für Systeme erster Ordnung angegeben werden, das auf dem globalen Arnoldi-Algorithmus basiert.

Sei $\mathbf{V}_{(k)}$ eine F -orthogonale Basis des Krylov-Unterraumes $\mathcal{K}_k(\mathbf{P}, \mathbf{Q})$, die mit dem globalen Arnoldi-Algorithmus generiert wurde, wobei \mathbf{P} und \mathbf{Q} Matrizen sind, die wie in Abschnitt 8.4.3 gewählt werden, um die Eigenschaft des Momentenabgleiches zu erfüllen. Dann führt die Anwendung der Projektion $\mathbf{\Pi} = \mathbf{V}_{(k)} \mathbf{V}_{(k)}^\dagger$ auf das System erster Ordnung der Form (3.4) zu einem reduzierten System erster Ordnung der Form (5.2) mit

$$\hat{\mathbf{E}} = \mathbf{V}_{(k)}^\dagger \mathbf{E} \mathbf{V}_{(k)}, \quad \hat{\mathbf{A}} = \mathbf{V}_{(k)}^\dagger \mathbf{A} \mathbf{V}_{(k)}, \quad \hat{\mathbf{B}} = \mathbf{V}_{(k)}^\dagger \mathbf{B}, \quad \text{und} \quad \hat{\mathbf{C}} = \mathbf{C} \mathbf{V}_{(k)}. \quad (10.28)$$

Da $\mathbf{V}_{(k)}$ F -orthonormal ist, muss anstatt $\mathbf{V}_{(k)}^*$ die Pseudo-Inverse

$$\mathbf{V}_{(k)}^\dagger = (\mathbf{V}_{(k)}^* \mathbf{V}_{(k)})^{-1} \mathbf{V}_{(k)}^* \quad (10.29)$$

für die obige Projektion benutzt werden, siehe Abschnitt 2.5.

Die Berechnung der Pseudoinversen $\mathbf{V}_{(k)}^\dagger$ kann numerisch instabil sein, aus diesem Grund wird in den hier vorgestellten Verfahren vor der Projektion des Systems eine \mathbf{QR} -Zerlegung $\mathbf{V}_{(k)} = \mathbf{V} \mathbf{R}$ berechnet, so dass $\mathbf{V}^* \mathbf{V} = \mathbf{I}_r$ gilt. Theoretisch erhält man damit

$$\mathbf{V}_{(k)}^\dagger = (\mathbf{V}_{(k)}^* \mathbf{V}_{(k)})^{-1} \mathbf{V}_{(k)}^* = ((\mathbf{V} \mathbf{R})^* \mathbf{V} \mathbf{R})^{-1} (\mathbf{V} \mathbf{R})^* = \mathbf{R}^{-1} \mathbf{V}^*. \quad (10.30)$$

Damit führt die Projektion $\mathbf{\Pi} = \mathbf{V}_{(k)} \mathbf{V}_{(k)}^\dagger$ zu einem reduzierten System der Form (5.2) mit

$$\hat{\mathbf{E}} = \mathbf{R}^{-1} \mathbf{V}^* \mathbf{E} \mathbf{V}, \quad \hat{\mathbf{A}} = \mathbf{R}^{-1} \mathbf{V}^* \mathbf{A} \mathbf{V}, \quad \hat{\mathbf{B}} = \mathbf{R}^{-1} \mathbf{V}^* \mathbf{B}, \quad \text{und} \quad \hat{\mathbf{C}} = \mathbf{C} \mathbf{V}. \quad (10.31)$$

Nach Multiplikation der ersten Gleichung in (5.2) von links mit \mathbf{R} erhält man somit das reduzierte System erster Ordnung der Form

$$\hat{\mathbf{E}} = \mathbf{V}^* \mathbf{E} \mathbf{V}, \quad \hat{\mathbf{A}} = \mathbf{V}^* \mathbf{A} \mathbf{V}, \quad \hat{\mathbf{B}} = \mathbf{V}^* \mathbf{B}, \quad \text{und} \quad \hat{\mathbf{C}} = \mathbf{C} \mathbf{V}. \quad (10.32)$$

In der Praxis führt somit die Projektion $\Pi = \mathbf{V}\mathbf{V}^*$ zu einem reduzierten System, das zu dem mit der Projektion $\Pi = \mathbf{V}_{(k)}\mathbf{V}_{(k)}^\dagger$ reduzierten System äquivalent ist, ohne dass die Matrix \mathbf{R}^{-1} benötigt wird.

Wird der globale Arnoldi-Algorithmus auf ein geeignetes Matrixpaar (\mathbf{P}, \mathbf{Q}) angewendet, dann behält die Eigenschaft des Momentenabgleiches, wie sie für den Block-Arnoldi-Algorithmus in Abschnitt 8.4.3 diskutiert wurde, ihre Gültigkeit. Die ersten Momente $\hat{\mathbf{H}}_j$ der Übertragungsfunktion des reduzierten Systems (10.28) sind dann die selben wie die ersten Momente \mathbf{H}_j der Übertragungsfunktion des Originalsystems (3.4). Der folgende Satz gibt diese enge Verbindung des Block-Arnoldi-Algorithmus und des globalen Arnoldi-Algorithmus bezüglich des reduzierten Systems wieder.

Satz 10.1 *Werden beim Block-Arnoldi-Algorithmus und beim globalen Arnoldi-Algorithmus die selben Matrizen zur Generierung von Basen der Krylov-Unterräume verwendet, dann sind die Übertragungsfunktionen der reduzierten Systeme, die durch die beiden Verfahren erzeugt werden, identisch.*

BEWEIS: Siehe [39]. □

10.4.2.2 Der globale Arnoldi-Algorithmus für Systeme zweiter Ordnung

Nachdem im vorhergehenden Abschnitt ein globaler Arnoldi-Algorithmus für Systeme erster Ordnung und sein Zusammenhang mit dem Block-Arnoldi-Algorithmus dargestellt wurden, lassen sich die gewonnenen Erkenntnisse auf Verfahren zur Reduktion von Systemen zweiter Ordnung übertragen. Damit kann ein neues Verfahren zur Reduktion von Systemen zweiter Ordnung basierend auf dem globalen Arnoldi-Algorithmus angegeben werden.

Angenommen $\mathbf{V}_{(k)}$ sei eine F -orthogonale Basis des Krylov-Unterraumes $\mathcal{K}_k(\mathbf{P}, \mathbf{Q})$, die mit dem globalen Arnoldi-Algorithmus generiert wurde, wobei \mathbf{P} und \mathbf{Q} Matrizen sind, die wie in Abschnitt 8.4.4 gewählt werden. Dann führt die Anwendung der Projektion $\Pi = \mathbf{V}_{(k)}\mathbf{V}_{(k)}^\dagger$ auf das System zweiter Ordnung (3.1) zu einem reduziertem System zweiter Ordnung der Form (5.4) mit

$$\begin{aligned}\hat{\mathbf{M}} &= \mathbf{V}_{(k)}^\dagger \mathbf{M} \mathbf{V}_{(k)}, & \hat{\mathbf{D}} &= \mathbf{V}_{(k)}^\dagger \mathbf{D} \mathbf{V}_{(k)}, & \hat{\mathbf{K}} &= \mathbf{V}_{(k)}^\dagger \mathbf{K} \mathbf{V}_{(k)}, \\ \hat{\mathbf{F}} &= \mathbf{V}_{(k)}^\dagger \mathbf{F}, & \hat{\mathbf{C}}_p &= \mathbf{C}_p \mathbf{V}_{(k)}, & \text{und} & \hat{\mathbf{C}}_v &= \mathbf{C}_v \mathbf{V}_{(k)}.\end{aligned}\tag{10.33}$$

Analog zum Vorgehen bei Verfahren für Systeme erster Ordnung lässt sich die Berechnung der Pseudoinversen vermeiden. Wird eine \mathbf{QR} -Zerlegung $\mathbf{V}_{(k)} = \mathbf{V}\mathbf{R}$ berechnet, so dass $\mathbf{V}^*\mathbf{V} = \mathbf{I}_r$, kann die Projektion $\Pi = \mathbf{V}\mathbf{V}^*$ angewendet werden, um ein reduziertes System zweiter Ordnung der Form

$$\begin{aligned}\hat{\mathbf{M}} &= \mathbf{V}^* \mathbf{M} \mathbf{V}, & \hat{\mathbf{D}} &= \mathbf{V}^* \mathbf{D} \mathbf{V}, & \hat{\mathbf{K}} &= \mathbf{V}^* \mathbf{K} \mathbf{V}, \\ \hat{\mathbf{F}} &= \mathbf{V}^* \mathbf{F}, & \hat{\mathbf{C}}_p &= \mathbf{V}^* \mathbf{C}_p \mathbf{V}, & \hat{\mathbf{C}}_v &= \mathbf{V}^* \mathbf{C}_v \mathbf{V}\end{aligned}\tag{10.34}$$

zu erhalten.

Werden der globale Arnoldi-Algorithmus mit einer geeigneten Wahl des Matrixpaares (\mathbf{P}, \mathbf{Q}) , wie sie in Abschnitt 8.4.4 beschrieben wurde, angewendet, dann behält die Eigenschaft des Momentenabgleiches, wie sie für den Block-Arnoldi-Algorithmus diskutiert wurde, ihre Gültigkeit.

Die ersten Momente $\hat{\mathbf{H}}_j$ der Übertragungsfunktion des reduzierten Systems (10.33) sind dann die selben wie die ersten Momente \mathbf{H}_j der Übertragungsfunktion des Originalsystems (3.1).

10.4.2.3 Der iterative rationale globale Arnoldi-Algorithmus für Systeme zweiter Ordnung (IRGA_2o)

Der im vorhergehenden Kapitel 10.3.2 beschriebene *modifizierte iterative rationale Krylov-Algorithmus* (MIRKA) kann mit dem globalen Arnoldi-Algorithmus zu einem effizienten Verfahren zur Reduktion von MIMO-Systemen kombiniert werden. Dadurch erhält man ein neues Verfahren, das die Reduktion eines MIMO-System zweiter Ordnung effizient berechnet und dabei die benötigten Entwicklungsstellen iterativ bestimmt. Dazu wird zur Berechnung der Projektionsmatrizen \mathbf{V} und ggf. \mathbf{W} in Zeile 6 in Algorithmus 16 der globale Arnoldi-Algorithmus als Krylov-Unterraum-Verfahren und die Projektion (10.34) verwendet, um das reduzierte System zweiter Ordnung zu erhalten.

Das aus der Kombination des *modifizierten iterativen rationalen Arnoldi-Algorithmus* für Systeme zweiter Ordnung (Algorithmus 19) und dem globalen Arnoldi-Algorithmus (Algorithmus 20) hervorgehende Verfahren, wird *iterativer rationaler globaler Arnoldi-Algorithmus für Systeme zweiter Ordnung*⁹ (IRGA_2o) genannt. Algorithmus 21 gibt das implementierte Verfahren in Pseudocode wieder.

Auch bei diesem Verfahren können prinzipiell alle Krylov-Unterraum-Verfahren für Systeme zweiter Ordnung verwendet werden. Wie beim Verfahren MIRKA_2o bereits dargestellt wurde, kann hier jedoch auf Verfahren basierend auf Krylov-Unterräumen zweiter Ordnung verzichtet werden, da in dieser Arbeit nur die Spezialfälle von ungedämpften und proportional gedämpften Systemen zweiter Ordnung betrachtet werden.

10.4.3 Untersuchung der benötigten Berechnungszeit

An dieser Stelle erfolgt ein Vergleiches der benötigten Berechnungszeiten der Verfahren MIRA_2o und IRGA_2o. Dazu wird das erste Beispielmmodell mit dem Verfahren IRGA_2o und dem Verfahren MIRA_2o reduziert. Um die Zunahme der Berechnungskosten bei zunehmender Anzahl von Eingängen des Systems zu untersuchen, wird die Berechnungszeit der verschiedenen Reduktionsverfahren anhand des ersten Beispielmmodells mit zunehmender Anzahl von Eingängen verglichen. Dazu wird das erste Beispielmmodell mit einer minimalen Anzahl von 2 Eingängen und einer maximalen Anzahl von 10 Eingängen sowie einer konstanten Anzahl von 6 Ausgängen verwendet. Da das erste Beispielmmodell eine Eingangsmatrix \mathbf{B} mit lediglich vier Eingangsvektoren besitzt, die für die Untersuchung zur Verfügung stehen, wird die Eingangsmatrix \mathbf{B} des Modells um bis zu sechs zufällig generierte Vektoren ergänzt. Das Verfahren MIRA_2o wird dabei in der Version mit einer blockweisen (MIRA_ND, MIRA_PD) (Algorithmus 19 mit Algorithmus 14 in Zeile 11) und in der Version mit einer spaltenweisen (MIRA_ND_sp, MIRA_PD_sp) (Algorithmus 19 mit Algorithmus 3 in Zeile 11) Berechnung der Projektionsmatrix \mathbf{V} durch den Arnoldi-Algorithmus verwendet und verglichen.

⁹In Englisch: *Iterative Rational Global Arnoldi Algorithm*

Algorithmus 21 Iterativer rationaler globaler Arnoldi-Algorithmus für Systeme zweiter Ordnung (IRGA_2o)

Eingabe: System Σ_{2o} mit Systemmatrizen M, D, K, F, C_p, C_v ; Vektor mit Anfangsentwicklungsstellen s ; Vektor mit Stufen der Krylov-Unterräume k ; Deflationstoleranz τ ; Toleranz $tol < 1$; Distanzparameter Δ ; maximale relevante Frequenz f_{max}

Ausgabe: Reduziertes System $\hat{\Sigma}_{2o}$ mit Systemmatrizen $\hat{M}, \hat{D}, \hat{K}, \hat{F}, \hat{C}_p, \hat{C}_v$ der Dimension $r = \text{columns}(F) \cdot \sum_{i=1}^{\text{length}(s)} (s)[k]_i$

```

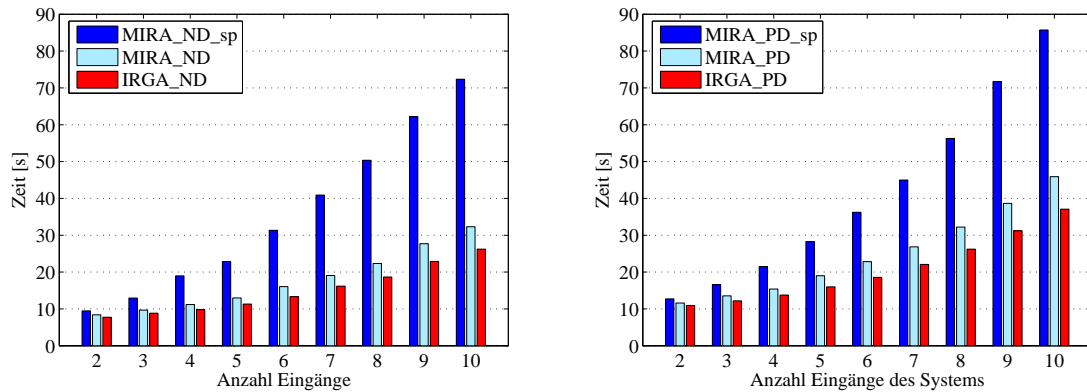
1: function [  $\hat{M}, \hat{D}, \hat{K}, \hat{F}, \hat{C}_p, \hat{C}_v$  ] = IRGA_2o(  $M, D, K, F, C_p, C_v, s, k, \tau, tol, \Delta, f_{max}$  )
2:  $\delta = 1$ ;  $R = 0$ ;  $iter = 0$ ;  $\hat{i} = \text{length}(s)$  ▷ Initialisierungen
3: Entfernung aller linear abhängigen Spaltenvektoren der Matrix  $F$ 
4: while  $\delta > tol$  ▷ Solange keine Konvergenz
5:    $iter = iter + 1$ 
6:    $V = [ ]$ ;  $W = [ ]$ 
7:   for  $i = 1 : \hat{i}$  ▷ Für jede Entwicklungsstelle
8:     Berechne die  $LU$ -Zerlegung:  $LU = ([s]_i^2 M + K)$  bzw.  $LU = ([s]_i^2 M + [s]_i D + K)$ 
9:      $P = -U \setminus (L \setminus M)$ 
10:     $Q = U \setminus (L \setminus F)$ 
11:     $V_i = \text{Globaler\_Arnoldi}(P, Q, [k]_i - 1)$ ; Algorithmus 20
12:  end for
13:   $\tilde{V} = [V_1 \ V_2 \ \dots \ V_{\hat{i}}]$ 
14:  Berechne die Rank-Revealing- $QR$ -Zerlegung:  $VR = \tilde{V}$ ,  $t = \text{rang}(\tilde{V}, \tau)$  ▷ (Re-) Orthonormalisierung
15:   $V = V(:, 1 : t)$ 
16:   $W = V$ 
17:  Berechne  $\hat{\Sigma}_{2o}$  mit  $\hat{M}, \hat{D}, \hat{K}, \hat{F}, \hat{C}_p, \hat{C}_v$  wie in (5.17) ▷ Generiere reduziertes System  $\hat{\Sigma}_{2o}$ 
18:   $[s, R] = \text{stellenauswahl}(\hat{\Sigma}_{2o}, s, R, \Delta, f_{max})$ ; Algorithmus 15 ▷ Neue Entwicklungsstellen
19:   $s = s(1 : \hat{i})$ 
20:   $\hat{\Sigma}_{2o, alt} \leftarrow \hat{\Sigma}_{2o}$ 
21:  if  $iter > 1$ 
22:    Berechne  $\delta = \|\hat{\Sigma}_{2o, alt} - \hat{\Sigma}_{2o}\|$  mit einer geeigneten Norm  $\|\cdot\|$ .
23:  end if
24: end while

```

Die Algorithmen der Reduktionsverfahren MIRA_2o (Algorithmus 19) und IRGA_2o (Algorithmus 21) werden mit den $\hat{i} = 6$ Anfangsentwicklungsstellen $i \cdot 2\pi$, $100i \cdot 2\pi$, $200i \cdot 2\pi$, $300i \cdot 2\pi$, $400i \cdot 2\pi$ und $500i \cdot 2\pi$ sowie $\tau = 10^{-30}$, $tol = 10^{-5}$ und $\Delta = (500/5) \cdot 2\pi$ verwendet.

Für die Algorithmen der Verfahren wird zudem $[k]_i = \hat{k} = 2$ für $i = 1, \dots, \hat{i}$ verwendet, womit man reduzierte Systeme der Dimension $r = m \cdot \hat{i} \cdot \hat{k}$ erhält. Die für den Vergleich reduzierten Systeme weisen somit eine Dimension auf, die zwischen 24 (Originalsystem mit zwei Eingängen) und 120 (Originalsystem mit zehn Eingängen) liegt. Unabhängig von der Anzahl der Eingänge konvergieren alle drei Varianten der Verfahren nach zwei Iterationen, wobei die Entwicklungsstellen aller drei Verfahren identisch sind. In Abbildung 10.9 sind die benötigten Berechnungszeiten graphisch dargestellt.

Der Vergleich der Berechnungszeiten der Verfahren basierend auf dem Block-Arnoldi und dem globalen Arnoldi-Algorithmus bestätigt die Aussagen der theoretischen Analyse der beiden Algorithmen. Der Block-Arnoldi-Algorithmus mit spaltenweiser Berechnung der Matrix V verursacht die größten numerischen Kosten aller hier betrachteten Varianten des rationalen Arnoldi-



(a) Verfahren mit Anwendung auf das ungedämpfte System (b) Verfahren mit Anwendung auf das proportional gedämpfte System

Abbildung 10.9: Benötigte Berechnungszeiten der Verfahren MIRA_2o und IRGA_2o - erstes Beispielmmodell

Verfahrens. Bedeutend kleiner ist der Unterschied der numerischen Kosten bei einem Vergleich der Verfahren basierend auf dem Block-Arnoldi-Algorithmus mit blockweiser Berechnung der Spaltenvektoren der Matrix V und dem Verfahren basierend auf dem globalen Arnoldi-Algorithmus. Dennoch zeigt sich, besonders bei zunehmender Anzahl an Eingängen, ein Vorteil zu Gunsten des globalen Arnoldi-Algorithmus.

10.4.4 Adaptive Modellreduktion

In den vorhergehenden Verfahren wurde die Stufe der Krylov-Unterräume zu jeder Entwicklungsstelle s_i mit $\hat{k} = 2$ fest gewählt. Mit dieser Wahl wurde erreicht, dass an jeder Entwicklungsstelle mindestens die ersten \hat{k} Momente des reduzierten Systems mit den ersten Momenten des Originalsystems übereinstimmen.

Anstatt diesen Ansatz weiter zu verfolgen, wird in diesem Abschnitt ein Verfahren entwickelt, das eine adaptive Methode zur Wahl der Stufe k_i des Krylov-Unterraumes zur jeweiligen Entwicklungsstelle s_i verwendet.

Die Erweiterung der Verfahren mit der iterativen Auswahl der Entwicklungsstellen, die im vorhergehenden Abschnitt vorgestellt wurden, resultiert in ein neues Reduktionsverfahren für Systeme zweiter Ordnung, das potenzielle Entwicklungsstellen iterativ und die Stufen der jeweiligen Krylov-Unterräume zu diesen Entwicklungsstellen adaptiv bestimmt.

10.4.4.1 Adaptive Wahl der Stufe der Krylov-Unterräume für Systeme erster Ordnung

In [39, 113] wurden Reduktionsverfahren entwickelt, die Systeme erster Ordnung mit adaptiver Wahl der Stufe k_i der Krylov-Unterräume zu jeder Entwicklungsstelle s_i reduzieren.

Das Verfahren, das in [39] vorgestellt wird, basiert auf dem in Abschnitt 9.3.2 vorgestellten Verfahren AORA und stellt eine Weiterentwicklung des Verfahrens für die Reduktion von MIMO-

Systemen erster Ordnung dar. Da das Verfahren zudem auf dem globalen Arnoldi-Algorithmus basiert, wurde das Verfahren *adaptiver rationaler globaler Arnoldi-Algorithmus*¹⁰ (AORGA) genannt.

Wie das Verfahren AORA bestimmt das Verfahren AORGA bei einer vorgegebenen Dimension des reduzierten Systems r die Stufen der Krylov-Unterräume k_i zu jeder Entwicklungsstelle s_i einer fest vorgegebenen Menge von potenziellen Entwicklungsstellen s_i , $i = 1, \dots, \hat{i}$ adaptiv. Diese Bestimmung der Stufen der Krylov-Unterräume k_i erfolgt durch die sukzessive Auswahl der Entwicklungsstellen s_i , mit deren Hilfe neue Spaltenvektoren, die zur Erweiterung der Projektionsmatrix \mathbf{V} dienen, berechnet werden. Dabei erfolgt die Auswahl der Entwicklungsstellen mit Hilfe des betragsmäßig größten j -ten Momentenfehlers (9.8) an den möglichen Entwicklungsstellen s_i .

Das mit diesem Algorithmus reduzierte System erzielt somit an den vorgegebenen Entwicklungsstellen die Reduzierung des größten Momentenfehlers aller möglichen reduzierten Systeme der gleichen Dimension, die auf Momentenabgleich basieren und die gleichen Entwicklungsstellen verwenden.

Diese Auswahlmethode führt dazu, dass Stellen, an denen die Übertragungsfunktion schwer zu approximieren ist und dadurch für viele Stufen der Krylov-Unterräume einen großen Momentenfehler besitzen, oft als Entwicklungsstellen verwendet werden, wohingegen Stellen, an denen die Übertragungsfunktion bereits gut approximiert wird und somit kleine Momentenfehler besitzen, kein einziges mal als Entwicklungsstelle verwendet werden. Wie beim Verfahren AORA führt diese Auswahl der Entwicklungsstellen im Ablauf des Reduktionsprozesses zu einer adaptiven Bestimmung der Stufen der Krylov-Unterräume, die zu den Entwicklungsstellen korrespondieren. Da das Verfahren auf dem globalen Arnoldi-Algorithmus basiert, erfolgt die Berechnung der Projektionsmatrix \mathbf{V} blockweise, d.h. besitzt die Eingangsmatrix des Systems $\mathbf{B} \tilde{m}$ unabhängige Spaltenvektoren, werden der Projektionsmatrix \mathbf{V} nach jeder Iteration des Algorithmus, falls keine linear abhängigen Vektoren während des Ablaufes des Algorithmus ausgelöscht werden, \tilde{m} Spaltenvektoren hinzugefügt.

Mit der adaptiven Bestimmung der Stufe des Krylov-Unterraumes zu einer Entwicklungsstelle wird somit auch dessen Dimension bestimmt. Die Dimension des reduzierten Systems sollte demnach mit r als dem Vielfachen der Anzahl der linear unabhängigen Eingänge \tilde{m} vorgegeben werden, damit keine bereits berechneten Spaltenvektoren verworfen werden müssen.

Der Algorithmus berechnet nach $J = r/\tilde{m}$ Iterationen eine Projektionsbasis mit $\mathbf{V} = [\mathbf{V}_1 \mathbf{V}_2 \dots \mathbf{V}_J]$. In der j -ten Iteration des Algorithmus wird zur Berechnung der Matrix \mathbf{V}_j eine Entwicklungsstelle aus der Menge der potenziellen Entwicklungsstellen gewählt, an der der maximale Momentenfehler vorhanden ist.

Mit Hilfe eines an den globalen Arnoldi-Algorithmus angepassten Satzes 9.3 aus Abschnitt 9.3.2 wird in [39, 42] der exakte Fehler des k -ten Momentes der Übertragungsfunktion eines mit dem globalen Arnoldi-Algorithmus und einer Entwicklungsstelle s_0 reduzierten Systems erster Ordnung analytisch hergeleitet.

¹⁰In Englisch: *Adaptive Order Rational Global Arnoldi Algorithm*

Satz 10.2 (k -ter Momentenfehler für Systeme erster Ordnung) *Resultiert ein reduziertes MIMO-System erster Ordnung aus der Anwendung der Projektion (10.28), wobei die ersten $j = 0, 1, \dots, k-1$ Momente des reduzierten Systems mit den ersten Momenten des Originalsystems an der Entwicklungsstelle s_0 zur Übereinstimmung gebracht werden und die Matrix $\mathbf{V}_{(k)} = [\mathbf{V}_0 \mathbf{V}_1 \dots \mathbf{V}_{k-1}]$ mit Hilfe des globalen Arnoldi-Algorithmus generiert wurde, dann kann der k -te Momentenfehler $\boldsymbol{\varepsilon}_k(s_0)$ an der Entwicklungsstelle s_0 als*

$$\boldsymbol{\varepsilon}_k(s_0) := \mathbf{H}_k(s_0) - \hat{\mathbf{H}}_k(s_0) = \mathbf{C} \left(\prod_{i=0}^k h_{i+1,i} \right) \mathbf{V}_k, \quad (10.35)$$

angegeben werden, wobei $\mathbf{H}_k(s_0)$ und $\hat{\mathbf{H}}_k(s_0)$ die k -ten Momente der Übertragungsfunktionen des unreduzierten bzw. des reduzierten Systems, $h_{1,0} = \| (s_0 \mathbf{E} - \mathbf{A})^{-1} \mathbf{B} \|_F$ und $h_{i+1,i}$ für $i = 1, \dots, k$ das $(i+1, i)$ -te Element der globalen Hessenbergmatrix $\mathbf{H}_{(k)}$ (siehe (10.24)) bezeichnet.

BEWEIS: Siehe [42]. □

Basierend auf Satz 10.2 lassen sich die Aussagen des Satzes 9.4 bezüglich der Berechnung der j -ten Momentenfehler $\boldsymbol{\varepsilon}_{j_i}(s_i)$ an den Entwicklungsstellen $s_i, i = 1, \dots, \hat{i}$ eines reduzierten Systems, das aus der Anwendung des Verfahrens AORA resultiert, auf reduzierte Systeme anwenden, die mit dem globalen Arnoldi-Algorithmus reduziert werden. Diese Vorgehensweise resultiert in den *adaptiven rationalen globalen Arnoldi-Algorithmus*¹¹ (AORGA), der in [39] näher dargestellt wird.

Für dieses Verfahren kann analog zu den Aussagen des Satzes 9.4 in der j -ten Iteration eine effiziente Berechnung der j -ten Momentenfehler $\boldsymbol{\varepsilon}_{j_i}(s_i)$ an den Entwicklungsstellen $s_i, i = 1, \dots, \hat{i}$ bezüglich der F -Norm angegeben werden:

$$\boldsymbol{\varepsilon}_{j_i}(s_i) = \|\mathbf{H}_{j_i}(s_i) - \hat{\mathbf{H}}_{j_i}(s_i)\|_F = \|h_\pi(s_i) \mathbf{C} \mathbf{R}^{(j-1)}(s_i)\|_F, \quad (10.36)$$

siehe dazu [39]. Dabei ist $h_\pi(s_i) = \prod_{t=1}^{j-1} \Gamma_t$, mit

$$\Gamma_t = \begin{cases} \|\mathbf{R}^{(t)}(s_i)\|, & \text{falls } s_i \text{ Entwicklungsstelle in der } t\text{-ten Iteration} \\ 1, & \text{sonst} \end{cases} \quad (10.37)$$

und $\mathbf{R}^{(j-1)}(s_i)$ Werte, die in der vorhergehenden $(j-1)$ -ten Iteration des Verfahrens AORGA, berechnet wurden.

Die Bestimmung der Momentenfehler an den potenziellen Entwicklungsstellen dient in der j -ten Iteration des Verfahrens dazu, die Entwicklungsstelle σ_j mit dem maximalen Momentenfehler

$$\sigma_j = \arg \max_{s_i} \|h_\pi(s_i) \mathbf{C} \mathbf{R}^{(j-1)}(s_i)\|_F \quad (10.38)$$

zu bestimmen. Demnach muss für jede vorgegebene Entwicklungsstelle s_i der Wert $h_\pi(s_i)$ und der Wert $\mathbf{C} \mathbf{R}^{(j-1)}(s_i)$ bestimmt werden, wobei der Wert $h_\pi(s_i)$ mit dem Wert 1 initialisiert und in jeder Iteration, in der die Entwicklungsstelle s_i ausgewählt wird mit

$$h_\pi(s_i) = h_\pi(s_i) \cdot \|\mathbf{R}^{(j-1)}(s_i)\|_F \quad (10.39)$$

aktualisiert wird.

¹¹In Englisch: *Adaptive Order Rational Global Arnoldi Algorithmus*

10.4.4.2 Adaptive Wahl der Stufe der Krylov-Unterräume für Systeme zweiter Ordnung

Die beschriebene Methode zur Bestimmung der Entwicklungsstelle mit dem maximalen Momentenfehler wird hier auf Verfahren zur Reduktion von Systemen zweiter Ordnung übertragen. Dazu wird analog zu Satz 10.2 eine Gleichung für den k -ten Momentenfehler ($k \geq 1$) für Systeme zweiter Ordnung hergeleitet werden. Der Beweis des folgenden Satzes orientiert sich dabei an der Vorgehensweise wie sie zum Beweis von Satz 2 in [42] dargestellt wird.

Satz 10.3 (k -ter Momentenfehler für Systeme zweiter Ordnung) *Wird ein MIMO-System zweiter Ordnung der Form (3.1) durch die Projektion (10.33) reduziert, wobei die ersten $j = 0, 1, 2, \dots, k-1$ Momente des reduzierten Systems an der Entwicklungsstelle s_0 mit den ersten Momenten des Originalsystems zur Übereinstimmung gebracht werden und die Matrix $\mathbf{V}_{(k)} = [\mathbf{V}_0 \mathbf{V}_1 \dots \mathbf{V}_{k-1}]$ mit Hilfe des globalen Arnoldi-Algorithmus generiert wurde, dann kann der k -te Momentenfehler $\boldsymbol{\varepsilon}_k(s_0)$ des reduzierten Systems zweiter Ordnung an der Entwicklungsstelle s_0 mit der Gleichung*

$$\boldsymbol{\varepsilon}_k(s_0) := \mathbf{H}_k(s_0) - \hat{\mathbf{H}}_k(s_0) = \mathbf{C}_v \left(\prod_{i=0}^{k-1} h_{i+1,i} \right) \mathbf{V}_{k-1} + (\mathbf{C}_p + s_0 \mathbf{C}_v) \left(\prod_{i=0}^k h_{i+1,i} \right) \mathbf{V}_k \quad (10.40)$$

ausgedrückt werden, wobei $\mathbf{H}_k(s_0)$ und $\hat{\mathbf{H}}_k(s_0)$ die k -ten Momente der Übertragungsfunktionen des unreduzierten bzw. des reduzierten Systems, $h_{1,0} = \|(s_0^2 \mathbf{M} + s_0 \mathbf{D} + \mathbf{K})^{-1} \mathbf{F}\|_F$ und $h_{i+1,i}$ für $i = 1, \dots, k$ das $(i+1, i)$ -te Element der globalen Hessenbergmatrix $\mathbf{H}_{(k)}$ bezeichnet.

BEWEIS: Seien $\mathbf{P} \in \mathbb{K}^{n \times n}$ und $\mathbf{Q} \in \mathbb{K}^{n \times s}$ die Matrizen, die mit dem globalen Arnoldi-Algorithmus zur Erzeugung der Projektionsmatrix $\mathbf{V}_{(k)}$ verwendet werden, wobei die Matrix \mathbf{Q} nur linear unabhängige Spaltenvektoren besitzt. Sei zudem $\tilde{\mathbf{H}}_{(k)} = \mathbf{H}_{(k)} \otimes \mathbf{I}_{\tilde{m}}$, wobei \tilde{m} die Anzahl der linear unabhängigen Spaltenvektoren der Matrix \mathbf{Q} bezeichnet. Dann kann die Gleichung (10.23) auch als

$$\mathbf{P} \mathbf{V}_{(k)} = \mathbf{V}_{(k)} \tilde{\mathbf{H}}_{(k)} + h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T, \quad (10.41)$$

mit $\mathbf{E}_k = \mathbf{e}_k \otimes \mathbf{I}_{\tilde{m}}$ geschrieben werden, wobei \mathbf{e}_k den k -ten Einheitsvektor geeigneter Dimension bezeichnet. Multipliziert man die Gleichung (10.41) von rechts mit $\mathbf{V}_{(k)}^\dagger$, dann erhält man die Gleichung

$$\mathbf{P} = \mathbf{V}_{(k)} \tilde{\mathbf{H}}_{(k)} \mathbf{V}_{(k)}^\dagger + h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T \mathbf{V}_{(k)}^\dagger, \quad (10.42)$$

Multipliziert man die Gleichung (10.41) $(k-1)$ -mal von links mit \mathbf{P} und mit \mathbf{E}_1 von rechts erhält man

$$\begin{aligned} \mathbf{P}^k \mathbf{V}_{(k)} \mathbf{E}_1 &= \mathbf{P}^{k-1} \mathbf{V}_{(k)} \tilde{\mathbf{H}}_{(k)} \mathbf{E}_1 + \mathbf{P}^{k-1} h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T \mathbf{E}_1 \\ &= \mathbf{V}_{(k)} \tilde{\mathbf{H}}_{(k)} \cancel{\mathbf{V}_{(k)}^\dagger \mathbf{V}_{(k)}} \tilde{\mathbf{H}}_{(k)} \cancel{\mathbf{V}_{(k)}^\dagger \mathbf{V}_{(k)}} \tilde{\mathbf{H}}_{(k)} \dots \tilde{\mathbf{H}}_{(k)} \cancel{\mathbf{V}_{(k)}^\dagger \mathbf{V}_{(k)}} \tilde{\mathbf{H}}_{(k)} \mathbf{E}_1 + \\ &\quad h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T \tilde{\mathbf{H}}_{(k)}^{k-1} \mathbf{E}_1 + \mathbf{P} h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T \tilde{\mathbf{H}}_{(k)}^{k-2} \mathbf{E}_1 + \mathbf{P}^2 h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T \tilde{\mathbf{H}}_{(k)}^{k-3} \mathbf{E}_1 + \\ &\quad \dots + \mathbf{P}^{k-2} h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T \tilde{\mathbf{H}}_{(k)} \mathbf{E}_1 + \mathbf{P}^{k-1} h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T \mathbf{E}_1 \end{aligned} \quad (10.43)$$

$$= \mathbf{V}_{(k)} \tilde{\mathbf{H}}_{(k)}^k \mathbf{E}_1 + \sum_{i=1}^k \mathbf{P}^{i-1} h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T \tilde{\mathbf{H}}_{(k)}^{k-i} \mathbf{E}_1. \quad (10.44)$$

Da $\mathbf{H}_{(k)}$ eine obere Hessenbergmatrix ist, gilt $\mathbf{E}_k^T \tilde{\mathbf{H}}_{(k)}^\ell \mathbf{E}_1 = \mathbf{0}$ für $\ell = 0, 1, 2, \dots, k-2$ und $\mathbf{E}_k^T \tilde{\mathbf{H}}_{(k)}^\ell \mathbf{E}_1 = \prod_{i=1}^{k-1} h_{i+1,i}$ für $\ell = k-1$. Durch Einsetzen in die Gleichung (10.43) erhält man somit

$$\begin{aligned} \mathbf{P}^k \mathbf{V}_{(k)} \mathbf{E}_1 &= \mathbf{V}_{(k)} \tilde{\mathbf{H}}_{(k)}^k \mathbf{E}_1 + \\ &\quad \underbrace{h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T \tilde{\mathbf{H}}_{(k)}^{k-1} \mathbf{E}_1}_{h_{k+1,k} \mathbf{V}_k \prod_{i=1}^{k-1} h_{i+1,i}} + \underbrace{P h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T \tilde{\mathbf{H}}_{(k)}^{k-2} \mathbf{E}_1}_0 + \underbrace{P^2 h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T \tilde{\mathbf{H}}_{(k)}^{k-3} \mathbf{E}_1}_0 + \\ &\quad \dots + \underbrace{P^{k-2} h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T \tilde{\mathbf{H}}_{(k)}^2 \mathbf{E}_1}_0 + \underbrace{P^{k-1} h_{k+1,k} \mathbf{V}_k \mathbf{E}_k^T \mathbf{E}_1}_0 \end{aligned} \quad (10.45)$$

$$= \mathbf{V}_{(k)} \tilde{\mathbf{H}}_{(k)}^k \mathbf{E}_1 + \left(\prod_{i=1}^k h_{i+1,i} \right) \mathbf{V}_k. \quad (10.46)$$

Mit Hilfe der Definition 3.12 der Momente von Systemen zweiter Ordnung an einer Entwicklungsstelle $\mathbf{s}_0 \in \mathbb{C}$ in Abschnitt 3.2 und mit der Projektion (10.33) kann der k -te ($k \geq 1$) Momentenfehler mit

$$\begin{aligned} \varepsilon_k(s_0) &:= \mathbf{H}_k(s_0) - \hat{\mathbf{H}}_k(s_0) \\ &= \left[\mathbf{C}_v \cdot \mathbf{P}^{k-1} \mathbf{Q} + (\mathbf{C}_p + s_0 \mathbf{C}_v) \cdot \mathbf{P}^k \mathbf{Q} \right] - \left[\mathbf{C}_v \mathbf{V}_{(k-1)} \cdot \mathbf{V}_{(k-1)}^\dagger \mathbf{P}^{k-1} \mathbf{V}_{(k-1)} \cdot \mathbf{V}_{(k-1)}^\dagger \mathbf{Q} \right. \\ &\quad \left. + (\mathbf{C}_p + s_0 \mathbf{C}_v) \mathbf{V}_{(k)} \cdot \mathbf{V}_{(k)}^\dagger \mathbf{P}^k \mathbf{V}_{(k)} \cdot \mathbf{V}_{(k-1)}^\dagger \mathbf{Q} \right] \end{aligned} \quad (10.47)$$

angegeben werden. Da $\mathbf{Q} = \mathbf{V}_{(k-1)} \mathbf{E}_1 h_{1,0}$ gilt, kann der Momentenfehler $\varepsilon_k(s_0)$ auch als

$$\begin{aligned} \varepsilon_k(s_0) &= \left[\mathbf{C}_v \cdot \mathbf{P}^{k-1} \mathbf{V}_{(k-1)} \mathbf{E}_1 h_{1,0} + (\mathbf{C}_p + s_0 \mathbf{C}_v) \cdot \mathbf{P}^k \mathbf{V}_{(k)} \mathbf{E}_1 h_{1,0} \right] - \\ &\quad \left[\mathbf{C}_v \mathbf{V}_{(k-1)} \cdot \mathbf{V}_{(k-1)}^\dagger \mathbf{P} \mathbf{V}_{(k-1)} \cdot \mathbf{V}_{(k-1)}^\dagger \mathbf{Q} + (\mathbf{C}_p + s_0 \mathbf{C}_v) \mathbf{V}_{(k)} \cdot \mathbf{V}_{(k)}^\dagger \mathbf{P} \mathbf{V}_{(k)} \cdot \mathbf{V}_{(k)}^\dagger \mathbf{Q} \right] \end{aligned} \quad (10.48)$$

geschrieben werden. Nach Umsortieren der Terme erhält man

$$\begin{aligned} \varepsilon_k(s_0) &= \mathbf{C}_v \left[(\mathbf{P}^{k-1} \mathbf{V}_{(k-1)} \mathbf{E}_1) h_{1,0} - \mathbf{V}_{(k-1)} \mathbf{V}_{(k-1)}^\dagger \mathbf{P} \mathbf{V}_{(k-1)} \mathbf{V}_{(k-1)}^\dagger \mathbf{Q} \right] \\ &\quad + (\mathbf{C}_p + s_0 \mathbf{C}_v) \left[(\mathbf{P}^k \mathbf{V}_{(k)} \mathbf{E}_1) h_{1,0} - \mathbf{V}_{(k)} \mathbf{V}_{(k)}^\dagger \mathbf{P} \mathbf{V}_{(k)} \mathbf{V}_{(k)}^\dagger \mathbf{Q} \right]. \end{aligned} \quad (10.49)$$

Durch Ersetzen von $\mathbf{P}^{k-1} \mathbf{V}_{(k-1)} \mathbf{E}_1$ und $\mathbf{P}^k \mathbf{V}_{(k)} \mathbf{E}_1$ in Gleichung (10.48) durch den Ausdruck

(10.45) erhält man:

$$\begin{aligned} \varepsilon_k(s_0) &= C_v \left[\left(V_{(k-1)} \tilde{H}_{(k-1)}^{k-1} E_1 + \prod_{i=1}^{k-1} h_{i+1,i} V_{k-1} \right) h_{1,0} - V_{(k-1)} V_{(k-1)}^\dagger P V_{(k-1)} V_{(k-1)}^\dagger Q \right] \\ &\quad + (C_p + s_0 C_v) \left[\left(V_{(k)} \tilde{H}_{(k)}^k E_1 + \prod_{i=1}^k h_{i+1,i} V_k \right) h_{1,0} - V_{(k)} V_{(k)}^\dagger P V_{(k)} V_{(k)}^\dagger Q \right] \end{aligned} \quad (10.50)$$

$$\begin{aligned} &= C_v V_{(k-1)} \tilde{H}_{(k-1)}^{k-1} E_1 h_{1,0} + C_v \left(\prod_{i=1}^{k-1} h_{i+1,i} V_{k-1} \right) h_{1,0} \\ &\quad - C_v V_{(k-1)} V_{(k-1)}^\dagger P V_{(k-1)} V_{(k-1)}^\dagger Q + (C_p + s_0 C_v) V_{(k)} \tilde{H}_{(k)}^k E_1 h_{1,0} \\ &\quad + (C_p + s_0 C_v) \left(\prod_{i=1}^k h_{i+1,i} V_k \right) h_{1,0} - (C_p + s_0 C_v) V_{(k)} V_{(k)}^\dagger P V_{(k)} V_{(k)}^\dagger Q. \end{aligned} \quad (10.51)$$

Da nach Voraussetzung das $k-1$ -te und das k -te Moment des reduzierten Systems mit denen des Originalsystems übereinstimmen, erhält man durch Kürzen des ersten und dritten sowie des vierten und sechsten Terms:

$$\begin{aligned} \varepsilon_k(s_0) &= \cancel{C_v V_{(k-1)} \tilde{H}_{(k-1)}^{k-1} E_1 h_{1,0}} + C_v \left(\prod_{i=1}^{k-1} h_{i+1,i} V_{k-1} \right) h_{1,0} \\ &\quad - \cancel{C_v V_{(k-1)} V_{(k-1)}^\dagger P V_{(k-1)} V_{(k-1)}^\dagger Q} + \cancel{(C_p + s_0 C_v) V_{(k)} \tilde{H}_{(k)}^k E_1 h_{1,0}} \\ &\quad + (C_p + s_0 C_v) \left(\prod_{i=1}^k h_{i+1,i} V_k \right) h_{1,0} - \cancel{(C_p + s_0 C_v) V_{(k)} V_{(k)}^\dagger P V_{(k)} V_{(k)}^\dagger Q} \end{aligned} \quad (10.52)$$

$$= C_v \left(\prod_{i=1}^{k-1} h_{i+1,i} \right) V_{k-1} h_{1,0} + (C_p + s_0 C_v) \left(\prod_{i=1}^k h_{i+1,i} \right) V_k h_{1,0} \quad (10.53)$$

$$= C_v \left(\prod_{i=0}^{k-1} h_{i+1,i} \right) V_{k-1} + (C_p + s_0 C_v) \left(\prod_{i=0}^k h_{i+1,i} \right) V_k. \quad (10.54)$$

□

Die Auswahl neuer Entwicklungsstellen erfolgt, dies wie beim Verfahren AORGA für Systeme erster Ordnung bereits beschrieben wurde, durch die Bestimmung der maximalen Momentenfehler. Die Berechnung der maximalen Momentenfehler erfolgt dabei analog dem Vorgehen bei Verfahren zur Reduktion von Systemen erster Ordnung mit den Aussagen, die der Satz 9.4 in Abschnitt 9.3 liefert. Demnach ist der j_i -ten Momentenfehler $\epsilon_{j_i}(s_i)$ der Übertragungsfunktion des reduzierten Systems an den potenziellen Entwicklungsstellen s_i , $i = 1, \dots, \hat{i}$ bezüglich der F -Norm in der j -ten Iteration des Verfahrens zur Reduktion von Systemen zweiter Ordnung über die Gleichung

$$\begin{aligned} \epsilon_{j_i}(s_i) &:= \|H_{j_i}(s_i) - \hat{H}_{j_i}(s_i)\|_F \\ &= \|h_{\pi}^{(j-2)}(s_i) C_v R^{(j-2)}(s_i) + h_{\pi}^{(j-1)}(s_i) (C_p + s_i C_v) R^{(j-1)}(s_i)\|_F \end{aligned} \quad (10.55)$$

zu bestimmen, wobei $\mathbf{H}_{j_i}(s_i)$ und $\hat{\mathbf{H}}_{j_i}(s_i)$ die j_i -ten Momente der Übertragungsfunktion des originalen bzw. des reduzierten Systems zweiter Ordnung bezeichnen. Die Werte $\mathbf{R}^{(j-1)}(s_i)$ und $\mathbf{R}^{(j-2)}(s_i)$ werden in der $(j-1)$ -ten bzw. $(j-2)$ -ten Iteration des Verfahrens und die Werte $h_\pi^{(j-2)}(s_i) = \prod_{t=1}^{j-2} \Gamma_t$ sowie $h_\pi^{(j-1)}(s_i) = \prod_{t=1}^{j-1} \Gamma_t$, werden mit

$$\Gamma_t = \begin{cases} \|\mathbf{R}^{(t)}(s_i)\|_F, & \text{falls } s_i \text{ Entwicklungsstelle in der } t\text{-ten Iteration} \\ 1, & \text{sonst} \end{cases} \quad (10.56)$$

berechnet.

10.4.4.3 Der adaptive iterative rationale globale Arnoldi-Algorithmus für Systeme zweiter Ordnung (AIRGA_2o)

In diesem Abschnitt wird ein neues Verfahren zur Reduktion von Systemen zweiter Ordnung der Form (3.1) vorgestellt, das auf einer iterativen Wahl potenzieller Entwicklungsstellen in Kombination mit einer adaptiven Wahl der Stufe der dazu korrespondierenden Krylov-Unterräume basiert.

Das neu entwickelte Verfahren geht dabei aus dem Verfahren IRGA_2o, der Methode zur iterativen Bestimmung der Entwicklungsstellen, wie sie im Verfahren MIRA_2o und der adaptiven Bestimmung der Stufe der Krylov-Unterräume zu den Entwicklungsstellen, wie sie im Verfahren AORGA verwendet wird, hervor.

Das kombinierte Verfahren wird im weiteren als *adaptiver iterativer rationaler globaler Arnoldi-Algorithmus*¹² für Systeme zweiter Ordnung (AIRGA_2o) genannt und zeichnet sich dadurch aus, dass es auf dem globalen Arnoldi-Algorithmus basiert und ein reduziertes System zweiter Ordnung durch iterative Bestimmung potenzieller Entwicklungsstellen in Verbindung mit einer adaptiven Bestimmung der Stufe der Krylov-Unterräume zu diesen Entwicklungsstellen berechnet.

Der Vorteil, der sich durch dieses Verfahrens gegenüber existierenden Verfahren ergibt, besteht somit darin, dass der Anwender von der Eingabe von Entwicklungsstellen und der Festlegung der Anzahl der Entwicklungen an den Entwicklungsstellen enthoben werden kann.

Dazu erfolgt im Algorithmus 22 des Verfahrens AIRGA_2o die Berechnung der Projektionsmatrix \mathbf{V} durch eine, zur Anwendung auf Systeme zweiter Ordnung angepasste, modifizierte Version der adaptiven Methode zur Bestimmung der Stufe der verwendeten Krylov-Unterräume des Verfahrens AORGA.

In der j -ten Iteration des Verfahrens AIRGA_2o wird in Zeile 14 des Algorithmus 22 die rechte Seite der Gleichung (10.55) verwendet, um die Entwicklungsstelle σ_j mit dem maximalen Momentenfehler

$$\sigma_j = \arg \max_{s_i} \|h_\pi^{(j-2)}(s_i) \mathbf{C}_v \mathbf{R}^{(j-2)}(s_i) + h_\pi^{(j-1)}(s_i) (\mathbf{C}_p + s_i \mathbf{C}_v) \mathbf{R}^{(j-1)}(s_i)\|_F \quad (10.57)$$

des reduzierten Systems zweiter Ordnung zu bestimmen. Dabei werden die Werte $\mathbf{R}^{(j-2)}(s_i)$ und $\mathbf{R}^{(j-1)}(s_i)$ in Zeile 28 bzw. 27 der vorhergehenden $(j-1)$ -ten Iteration berechnet. Der Wert

¹²In Englisch: *Adaptive Order Iterative Rational Global Arnoldi Algorithmus*

$h_{\pi}^{(0)}(s_i)$ wird mit 1 initialisiert und bei jeder Iteration, in der die Stelle s_i als Entwicklungsstelle ausgewählt wird, mit

$$h_{\pi}^{(j)}(s_i) = h_{\pi}^{(j-1)}(s_i) \cdot \|\mathbf{R}^{(j-1)}(s_i)\|_F \quad (10.58)$$

aktualisiert (Zeile 20).

Da die Wahl der Stufe des Krylov-Unterraumes an jeder Entwicklungsstelle s_i durch eine adaptive Methode stattfindet, ist die Anzahl der Momente des reduzierten Systems, die mit den ersten Momenten der Übertragungsfunktion des originalen Systems an den Entwicklungstellen zur Übereinstimmung gebracht werden, unterschiedlich hoch.

Im Gegensatz zu den vorhergehenden Verfahren können somit a priori keine Aussagen über die Anzahl der Momente, die an jeder Entwicklungsstelle zur Übereinstimmung gebracht werden, gemacht werden.

Der Algorithmus des entwickelten Verfahrens ist als Algorithmus 22 in Pseudocode dargestellt und die Abbildung 10.10 gibt den Ablauf dieses Algorithmus in einem Ablaufdiagramm wieder.

10.4.4.4 Der vollständige adaptive iterative rationale globale Arnoldi-Algorithmus für Systeme zweiter Ordnung (VAIRGA_2o)

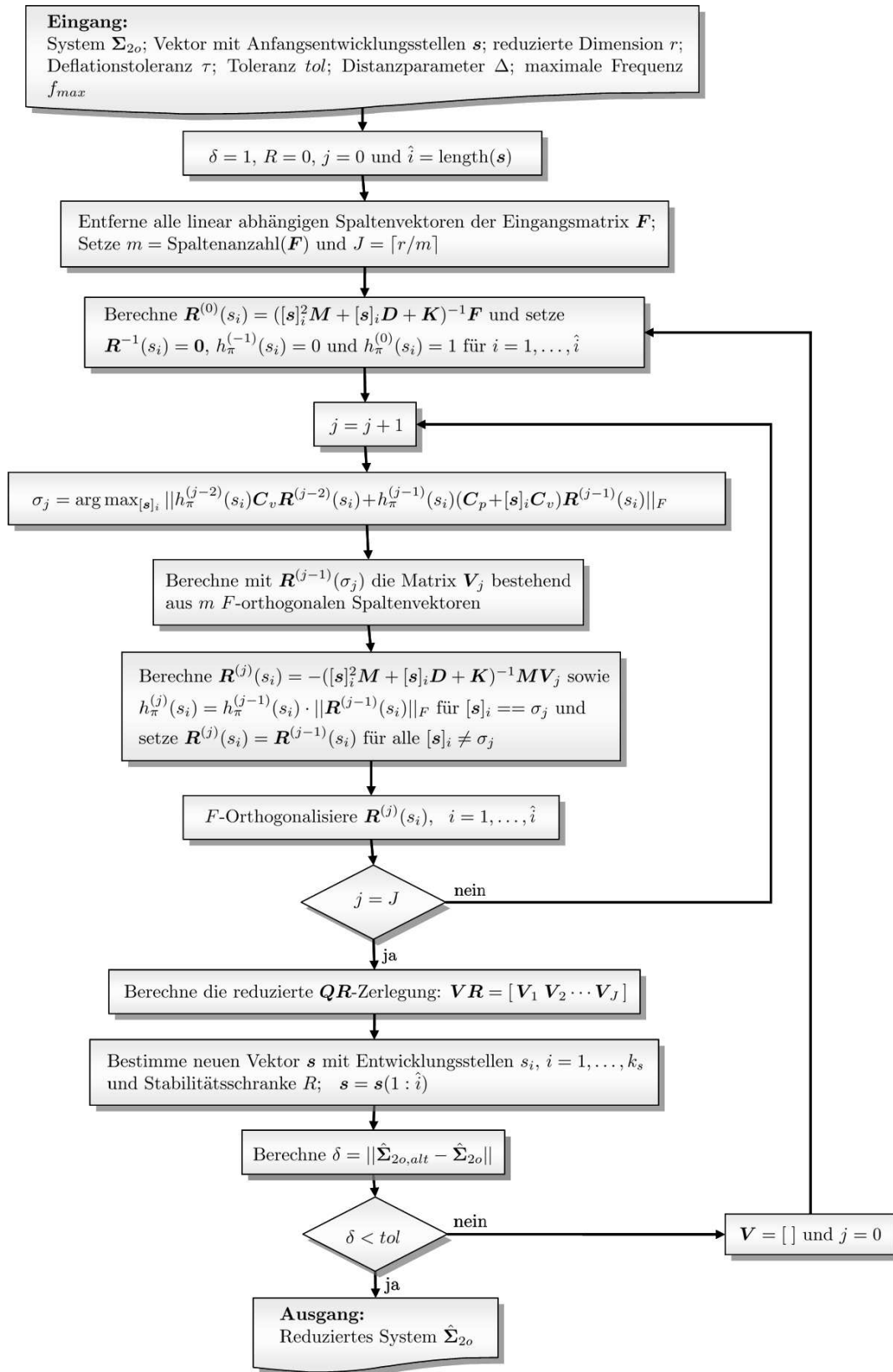
Das Verfahren AIRGA_2o verwendet die im vorhergehenden Abschnitt 10.3.2 vorgestellte Methode zur Bestimmung der Entwicklungsstellen. Bei dieser Methode werden die Eigenwerte des reduzierten Systems in einem vorgegebenen Frequenzbereich zur iterativen Bestimmung neuer Entwicklungsstellen herangezogen. Um eine Häufung von Entwicklungsstellen an einer Stelle des Frequenzbereiches zu vermeiden wurde ein Parameter Δ verwendet, der es ermöglicht nur solche Eigenwerte zu betrachten, die eine bestimmte Distanz voneinander aufweisen. Dadurch wird sicher gestellt, dass die neuen Entwicklungsstellen über den Frequenzbereich verteilt liegen.

Da die Wahl des Wertes des Parameters mit $\Delta = (\text{maximale relevante Frequenz} \cdot 2\pi) / (\text{Anzahl Anfangsentwicklungsstellen} - 1) = (2\pi \cdot f_{max}) / (\hat{i} - 1)$ intuitiv erfolgt, wird an dieser Stelle eine Variante des Verfahrens AIRGA_2o vorgestellt, die auf die Verwendung einer festgelegten minimalen Distanz verzichtet.

Bei dieser Variante des Verfahren AIRGA_2o wird der Parameter Δ in Algorithmus 22 mit $\Delta = 0$ gewählt, was zur Folge hat, dass der Algorithmus *Stellenauswahl* (Algorithmus 15) für die folgende Reduktions-Iteration alle Eigenwerte des reduzierten Systems im vorgegebenen Frequenzbereich zur Bestimmung neuer Entwicklungsstellen verwendet.

Zudem wird die Anzahl der potenziellen Entwicklungsstellen der folgenden Reduktions-Iteration nicht auf \hat{i} Stellen beschränkt (Algorithmus 22, Zeile 38 entfällt), wodurch der folgenden Reduktions-Iteration, wenn genügend Eigenwerte vorhanden sind, somit $k_s \geq \hat{i}$ potenzielle Entwicklungsstellen der adaptiven Bestimmung der Stufen der dazu korrespondierenden Krylov-Unterräume zur Verfügung stehen. Dadurch ist es möglich, dass die Anzahl der potenziellen Entwicklungsstellen mit zunehmender Anzahl der Reduktions-Iterationen ansteigt.

Das Verfahren wird wegen der Verwendung der vollständigen Menge der im vorgegebenen Frequenzbereich liegenden Eigenwerte zur Bestimmung der Entwicklungsstellen der folgenden Itera-


 Abbildung 10.10: Ablaufdiagramm des *Adaptiven iterativen rationalen globalen Arnoldi-Algorithmus* für Systeme zweiter Ordnung (AIRGA_2o)

Algorithmus 22 Adaptiver iterativer rationaler globaler Arnoldi-Algorithmus für Systeme zweiter Ordnung (AIRGA_2o)

Eingabe: System Σ_{2o} mit Matrizen M, D, K, F, C_p, C_v ; Vektor mit Anfangsentwicklungsstellen s ; reduzierte Dimension r ; Deflationstoleranz τ ; Toleranz $tol < 1$; Distanzparameter Δ ; maximale relevante Frequenz f_{max}

Ausgabe: Reduziertes System $\hat{\Sigma}_{2o}$ mit Systemmatrizen $\hat{M}, \hat{D}, \hat{K}, \hat{F}, \hat{C}_p, \hat{C}_v$; Sequenz der verwendeten Entwicklungsstellen $\sigma_j, j = 1, \dots, \lceil r/m \rceil$

```

1: function  $[\hat{M}, \hat{D}, \hat{K}, \hat{F}, \hat{C}_p, \hat{C}_v, k_i, \sigma_i] = \text{AIRGA\_2o}(M, D, K, F, C_p, C_v, s, r, \tau, tol, \Delta, f_{max})$ 
2:  $\delta = 1$ ;  $iter = 0$ ;  $\hat{i} = \text{length}(s)$ ;  $R = 0$ ;  $\Delta = 1$  ▷ Initialisierungen
3: Entferne alle linear abhängigen Spaltenvektoren der Matrix  $F$ 
4:  $\tilde{m} = \text{Spaltenanzahl}(F)$ 
5:  $J := \lceil r/\tilde{m} \rceil$  ▷ Anzahl der Global-Arnoldi-Iterationen
6: while  $\delta > tol$  ▷ Reduktions-Iterationen
7:    $iter = iter + 1$ 
8:    $V = [ ]$  ▷ (Re-)Initialisierung
9:   for  $i = 1 : \text{length}(s)$ 
10:      $R^{-1}(s_i) = 0$ ;  $h_{\pi}^{(-1)}(s_i) = 0$ 
11:      $R^{(0)}(s_i) := (s_i^2 M + s_i D + K)^{-1} F$ ;  $h_{\pi}^{(0)}(s_i) := 1$  ▷ Residuum des 0. Momentes bei  $s_i$ 
12:   end for
13:   for  $j = 1, 2, \dots, J$  ▷ Global-Arnoldi-Iterationen
14:      $\sigma_j = \arg \max_{s_i} \|h_{\pi}^{(j-2)}(s_i) C_v R^{(j-2)}(s_i) + h_{\pi}^{(j-1)}(s_i) (C_p + s_i C_v) R^{(j-1)}(s_i)\|_F$  ▷ Auswahl von  $\sigma_j$ 
15:      $h_{j,j-1}(\sigma_j) := \|R^{(j-1)}(\sigma_j)\|_F$ 
16:      $V_j := R^{(j-1)}(\sigma_j)/h_{j,j-1}(\sigma_j)$  ▷  $F$ -Orthonormalisierung der Spalten von  $V_j$ 
17:     for  $i = 1 : \text{length}(s)$  ▷ Berechne die Residuen  $R^{(j)}(s_i)$  für die nächste Iteration
18:       if  $(s_i == \sigma_j)$ 
19:          $Z = -(s_i^2 M + s_i D + K)^{-1} M V_j$ 
20:          $h_{\pi}^{(j)}(s_i) = h_{\pi}^{(j-1)}(s_i) \cdot h_{j,j-1}(\sigma_j)$ 
21:         Berechne die Rank-Revealing- $QR$ -Zerlegung:  $W R = Z$ ;  $t = \text{rang}(Z, \tau)$ 
22:          $R^{(j)}(s_i) = W(:, 1:t)$ 
23:       else
24:          $R^{(j)}(s_i) = R^{(j-1)}(s_i)$ ;  $h_{\pi}^{(j)}(s_i) = h_{\pi}^{(j-1)}(s_i)$ 
25:       end if
26:     for  $t = 1, 2, \dots, j$  ▷  $F$ -Orthogonalisierung
27:        $u = \text{columns}(R^{(j)}(s_i))$ ;  $\gamma_{t,j}(s_i) = \text{spur}((V_t(:, 1:u))^* R^{(j)}(s_i))$ 
28:        $v = \text{columns}(R^{(j-1)}(s_i))$ ;  $\delta_{t,j}(s_i) = \text{spur}((V_t(:, 1:v))^* R^{(j-1)}(s_i))$ 
29:        $R^{(j)}(s_i) = R^{(j)}(s_i) - \gamma_{t,j}(s_i) V_t(:, 1:u)$ ;  $R^{(j-1)}(s_i) = R^{(j-1)}(s_i) - \delta_{t,j}(s_i) V_t(:, 1:v)$ 
30:     end for
31:   end for
32:   end for
33:    $\tilde{V} = [V_1 V_2 \dots V_J]$  ▷ Erstellen der Matrix  $V$ 
34:   Berechne die Rank-Revealing- $QR$ -Zerlegung:  $\tilde{V} R = \tilde{V}$ ;  $t = \text{rang}(\tilde{V}, \tau)$ 
35:    $V = \tilde{V}(:, 1 : \min\{r, t\})$  ▷ Dimensionsanpassung, wenn  $r$  kein Vielfaches von  $m$ 
36:   Berechne  $\hat{\Sigma}_{2o}$  mit  $\hat{M}, \hat{D}, \hat{K}, \hat{F}, \hat{C}_p, \hat{C}_v$  wie in (5.17) ▷ Generiere reduziertes System  $\hat{\Sigma}_{2o}$ 
37:    $[s, R] = \text{stellenauswahl}(\hat{\Sigma}_{2o}, s, R, \Delta, f_{max})$ ; Algorithmus 15
38:    $s = s(1 : \hat{i})$ 
39:    $\hat{\Sigma}_{2o,alt} \leftarrow \hat{\Sigma}_{2o}$ 
40:   if  $iter > 1$ 
41:     Berechne  $\delta = \|\hat{\Sigma}_{2o,alt} - \hat{\Sigma}_{2o}\|$  mit einer geeigneten Norm  $\|\cdot\|$ 
42:   end if
43: end while

```

tion vollständiger aadaptiver iterativer rationaler globaler Arnoldi-Algorithmus für Systeme zweiter Ordnung (VAIRGA_2o) genannt.

10.4.4.5 Diskussion der Approximationseigenschaften

Um die Eigenschaften des in diesem Kapitel entwickelten adaptiven Verfahren AIRGA_2o und VAIRGA_2o mit den beiden bereits vorgestellten Verfahren MIRA_2o und IRGA_2o zu untersuchen, werden die vier Verfahren auf die beiden vorgestellten Beispielm Modelle angewendet.

Der Vergleich erfolgt bezüglich der relativen Approximationsfehler (10.3) und (10.5), wobei das Frequenzintervall $[0 \text{ Hz}, 500 \text{ Hz}]$ als relevantes Intervall betrachtet wird.

Dabei werden die Krylov-Unterraum-Verfahren, wie bereits im vorhergehenden Abschnitt, auch auf Unterschiede in den Approximationsergebnissen bei Anwendung auf ungedämpfte Systeme mit anschließender proportionaler Dämpfung mit den reduzierten Systemmatrizen und der direkten Anwendung auf proportional gedämpfte Systeme untersucht.

Die reduzierten Systeme bei Anwendung der Krylov-Unterraum-Verfahren auf ungedämpfte Systeme werden MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND genannt, wohingegen die reduzierten Systeme bei Anwendung der Krylov-Unterraum-Algorithmen auf proportional gedämpfte Systeme MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD genannt werden.

Erstes Beispielm Modell

Zunächst erfolgt die Betrachtung der Approximationsergebnisse der reduzierten Systeme anhand des ersten Beispielm Modells der Dimension 4983, das vier Eingänge und sechs Ausgänge besitzt. Dazu erfolgt ein Vergleich der Approximationsergebnisse der mit den verschiedenen Verfahren reduzierten Systeme.

In Anhang A.1 sind die Approximationsfehler des mit den verschiedenen Verfahren reduzierten ersten Beispielm Modells der Dimension 16, 24, 32 und 40 dargestellt. Zur ausführlichen Darstellung der Approximationsergebnisse wird hier die reduzierte Dimension 32 gewählt, da diese Dimension mit den Dimensionen reduzierter Systeme, die aus der Anwendung etablierter Reduktionsverfahren auf Modelle dieser Größenordnung resultieren, vergleichbar ist und die reduzierten Systeme mit dieser Dimension im gewählten Frequenzbereich bereits sehr gute Approximationsergebnisse liefern.

Die Krylov-Unterraum reduzierten Modelle mit reduzierter Dimension 32 werden dazu einem modal reduzierten Modell der gleichen Dimension (MODAL_2o) gegenübergestellt, welches mit dem FEM-Programm MSC.Nastran[®] generiert wird. Diese Dimension des modal reduzierten Systems entspricht dabei der Anzahl der Eigenmoden im Intervall $[0 \text{ Hz}, 1464.1 \text{ Hz}]$.

Die Algorithmen der Reduktionsverfahren (Algorithmus 19, Algorithmus 21 und Algorithmus 22) werden mit den Anfangsentwicklungsstellen $i \cdot 2\pi$, $166i \cdot 2\pi$, $332i \cdot 2\pi$ und $500i \cdot 2\pi$ sowie $\tau = 10^{-30}$, $tol = 10^{-5}$, $f_{max} = 500$ und $\Delta = (500/3) \cdot 2\pi$ (AIRGA_2o) bzw. $\Delta = 0$ (VAIRGA_2o) verwendet. Die Algorithmen des Verfahrens MIRA_2o (Algorithmus 19) und des Verfahrens IRGA_2o (Algorithmus 21) verwenden für die Stufe des Krylov-Unterraumes $k_i = [\mathbf{k}]_i = 2$ an jeder Entwicklungsstelle $s_i = [\mathbf{s}]_i$, $i = 1, \dots, \hat{i}$. Die Dimension der reduzierten Systeme r ergibt

sich durch die Anzahl der Anfangsentwicklungsstellen \hat{i} , der Stufe des Krylov-Unterraumes k_i an jeder Entwicklungsstelle s_i und der Anzahl der (linear unabhängigen) Eingänge $\tilde{m} = 4$ damit zu $r = \tilde{m} \cdot \hat{i} \cdot \hat{k} = 4 \cdot 4 \cdot 2 = 32$.

Die Ergebnisse der verschiedenen Reduktionsverfahren sind in der Tabelle 10.3 und die verwendeten Entwicklungsstellen in der letzten Iteration der Verfahren und deren Verwendungsreihenfolge sind in der Tabelle 10.4 bzw. Tabelle 10.5 zusammengefasst. Da die Berechnung des modal reduzierten Systems mit des FEM-Programms MSC.Nastran[®] erfolgt, die bezüglich der Berechnungszeit und des Speicherbedarfs optimale Algorithmen verwendet, wird hier auf den Vergleich der Berechnungszeit des modal reduzierten Systems mit den übrigen reduzierten Systemen verzichtet.

Die Abbildungen 10.11 und 10.13 geben den Verlauf der relativen Fehler (10.3) und (10.5) der verschiedenen reduzierten Modelle wieder. In den Abbildungen 10.12 und 10.14 ist ein Vergleich der Verfahren AIRGA_ND und VAIRGA_ND bzw. AIRGA_PD und VAIRGA_PD dargestellt.

Tabelle 10.3: Ergebnisse der untersuchten Reduktionsverfahren - erstes Beispielmmodell

Verfahren	Moden	Zeit [s]	Anzahl Iterationen	Fehler $\epsilon_{max}(\mathcal{I})$ (10.4)	Fehler ϵ_{max} (10.6)	Fehler ϵ_{rel} (10.7)
MODAL_2o	32	–	2	20.56	$3.84 \cdot 10^{-3}$	$14.5 \cdot 10^{-3}$
Verfahren mit Anwendung auf das ungedämpfte System						
MIRA_ND	–	6.25	2	$11.82 \cdot 10^{-3}$	$0.86 \cdot 10^{-6}$	$11.9 \cdot 10^{-3}$
IRGA_ND	–	5.64	2	$0.24 \cdot 10^{-3}$	$17.06 \cdot 10^{-6}$	$10.8 \cdot 10^{-3}$
AIRGA_ND	–	9.97	2	$10.89 \cdot 10^{-3}$	$9.73 \cdot 10^{-6}$	$14 \cdot 10^{-3}$
VAIRGA_ND	–	23.13	2	$3.63 \cdot 10^{-3}$	$2.44 \cdot 10^{-6}$	$13.1 \cdot 10^{-3}$
Verfahren mit Anwendung auf das proportional gedämpfte System						
MIRA_PD	–	9.05	2	$6.14 \cdot 10^{-3}$	$2.13 \cdot 10^{-6}$	$9.4 \cdot 10^{-3}$
IRGA_PD	–	8.25	2	$0.47 \cdot 10^{-3}$	$10.69 \cdot 10^{-6}$	$9.5 \cdot 10^{-3}$
AIRGA_PD	–	14.61	2	$24.73 \cdot 10^{-3}$	$21.81 \cdot 10^{-6}$	$12.1 \cdot 10^{-3}$
VAIRGA_PD	–	27.95	2	$0.67 \cdot 10^{-3}$	$2.86 \cdot 10^{-6}$	$12.1 \cdot 10^{-3}$

Tabelle 10.4: Entwicklungsstellen der letzten Iteration der untersuchten Reduktionsverfahren - erstes Beispielmmodell

Nr.	Entwicklungsstelle	Nr.	Entwicklungsstelle	Nr.	Entwicklungsstelle
1	$1.59 \cdot 10^{-3} + 0.11 \imath$	2	$0.24 + 75.68 \imath$	3	$0.38 + 95.50 \imath$
4	$0.39 + 95.81 \imath$	5	$0.39 + 96.00 \imath$	6	$1.19 + 168.28 \imath$
7	$1.29 + 175.09 \imath$	8	$2.70 + 253.68 \imath$	9	$10.38 + 497.59 \imath$

Beim Vergleich der Fehlerverläufe in Abbildung 10.11 und Abbildung 10.13 mit dem modal re-

Tabelle 10.5: Reihenfolge der Verwendung der Entwicklungsstellen in der letzten Iteration der untersuchten Reduktionsverfahren - erstes Beispielmodell

Verfahren	Reihenfolge der verwendeten Entwicklungsstellen
MIRA_ND, IRGA_ND MIRA_PD, IRGA_PD	1, 1, 9, 9, 2, 2, 8, 8
AIRGA_ND, AIRGA_PD	1, 9, 2, 1, 8, 2, 8, 1
VAIRGA_ND	1, 9, 5, 3, 2, 6, 4, 7
VAIRGA_PD	1, 9, 5, 3, 2, 6, 7, 8

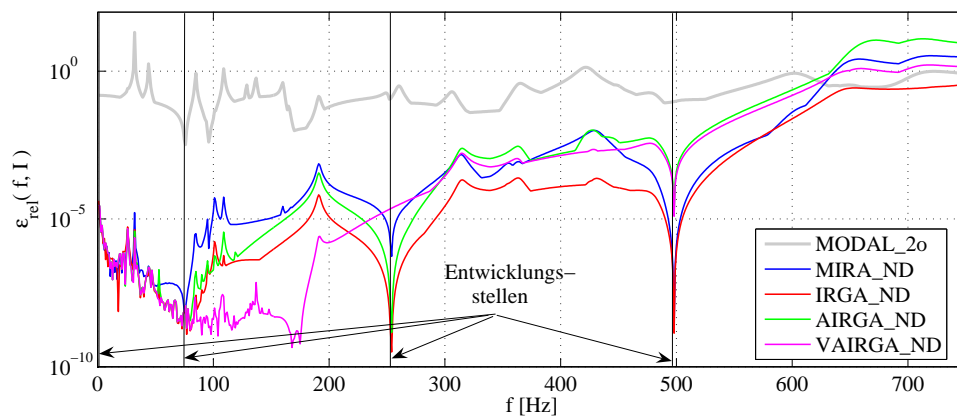
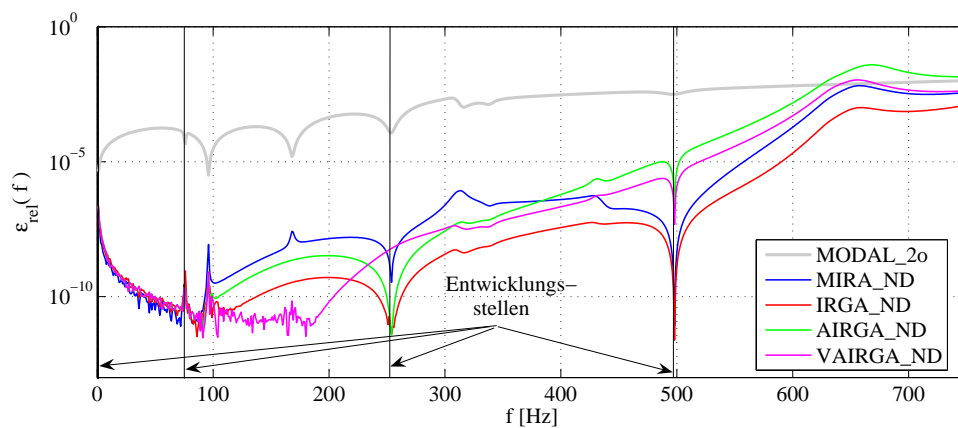

 (a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$

 (b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung 10.11: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND - erstes Beispielmodell

duzierten Modell MODAL_2o, zeigt sich für alle Krylov-Unterraum-Verfahren eine durchgehend deutlich bessere Approximation des relevanten Frequenzbereiches bis 500 Hz. Das Verfahren AIRGA_2o weist sogar im gesamten relevanten Frequenzbereich eine bessere Approximation als das modal reduzierte Modell auf.

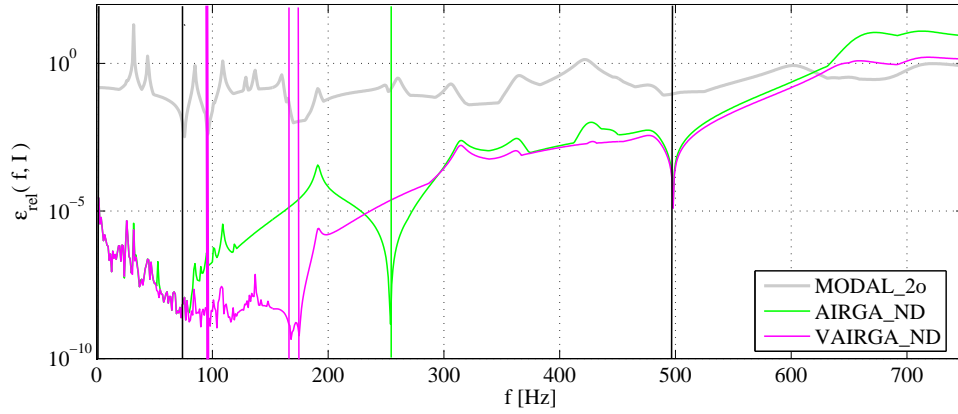
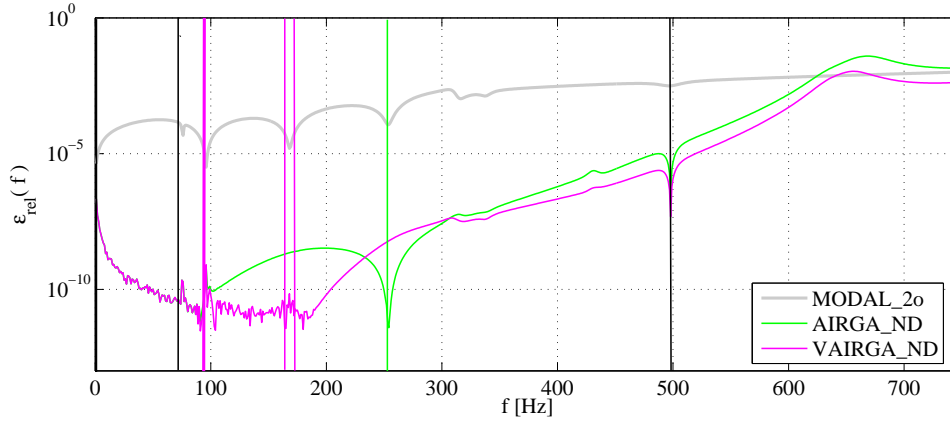
(a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$ (b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung 10.12: Relative Fehler der reduzierten Systeme MODAL_2o, AIRGA_ND und VAIRGA_ND mit verwendeten Entwicklungsstellen in der letzten Iteration der Krylov-Unterraum-Verfahren (grün: Entwicklungsstelle des Verfahrens AIRGA_ND, magenta: Entwicklungsstellen des Verfahrens VAIRGA_ND, schwarz: Entwicklungsstellen beider Verfahren) - erstes Beispielmodell

Bei einem Vergleich der Ergebnisse der vier in dieser Arbeit vorgestellten Krylov-Unterraum-Verfahren bezüglich des relativen Fehlers $\epsilon_{rel}(f)$ zeigt sich ein Vorteil des Verfahrens VAIRGA_2o gegenüber den übrigen Verfahren.

Die mit den Verfahren MIRA_2o, IRGA_2o und AIRGA_2o reduzierten Systeme zeigen ab 96 Hz eine deutliche Verringerung der Approximationsgüte, wohingegen die reduzierten Systeme VAIRGA_ND und VAIRGA_PD das Originalsystem bis zu einer Frequenz von 235 Hz besser approximieren.

Der Grund dafür ist in der Verwendung aller Eigenwerte des in der vorhergehenden Iteration reduzierten Systems im relevanten Frequenzbereich zur Bestimmung potenzieller Entwicklungsstellen in der aktuellen Iteration und der adaptiven Bestimmung geeigneter Entwicklungsstellen aus der Menge dieser Eigenwerte zu sehen. Allerdings steigt der Approximationsfehler der mit diesem Verfahren reduzierten Systeme bei höheren Frequenzen schnell an, was durch die Konzen-

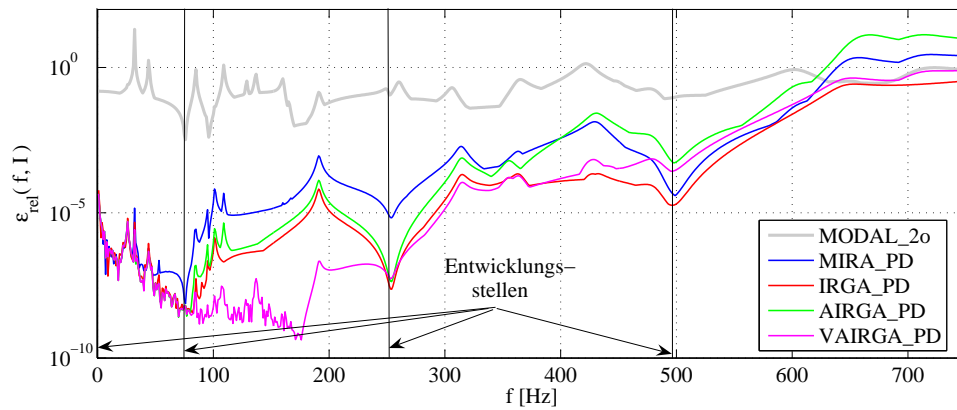
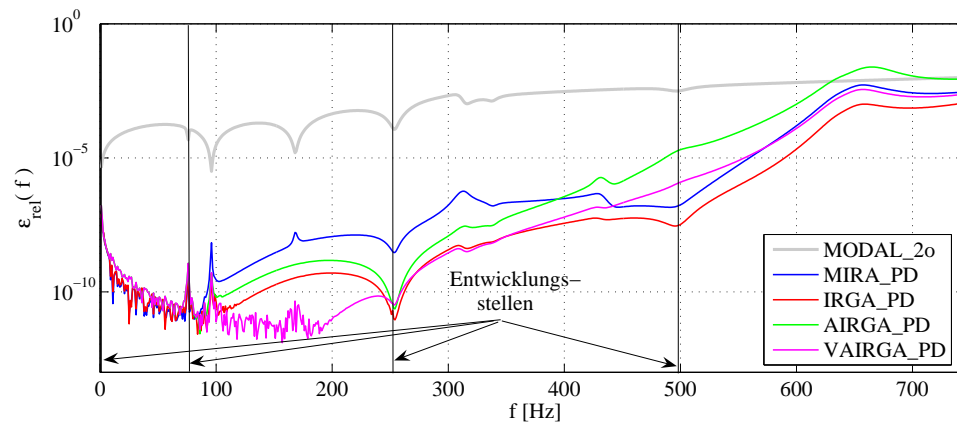

 (a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$

 (b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung 10.13: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD - erstes Beispielmmodell

tration der adaptiv ausgewählten Entwicklungsstellen im niederfrequenten Bereich zu erklären ist. Zudem benötigt das Verfahren VAIRGA_2o wegen der höheren Anzahl von potenziellen Entwicklungsstellen und der durchzuführenden adaptiven Bestimmung neuer Entwicklungsstellen deutlich mehr Berechnungszeit als die übrigen Verfahren, die nur eine beschränkte Anzahl von Eigenwerten im relevanten Frequenzintervall zur Bestimmung neuer potenzieller Entwicklungsstellen verwenden.

Bei einem Vergleich der Ergebnisse der reduzierten Systeme, die aus der Anwendung der Verfahren MIRA und IRGA auf das ungedämpfte bzw. das proportional gedämpfte System resultieren, zeigen sich, bis auf eine etwas bessere Approximation an den Entwicklungsstellen der reduzierten Systeme MIRA_ND und IRGA_ND, annähernd identische Approximationsergebnisse.

Bei Betrachtung der reduzierten Systeme AIRGA_ND und AIRGA_PD sowie VAIRGA_ND und VAIRGA_PD, die aus der Anwendung des Verfahrens AIRGA_2o bzw. VAIRGA_2o auf das ungedämpfte bzw. das proportional gedämpfte System erzielt werden, zeigt das aus der Anwendung auf das proportional gedämpfte System resultierende reduzierte System AIRGA_PD bessere Approximationsergebnisse als das aus der Anwendung auf das ungedämpfte System resultierende

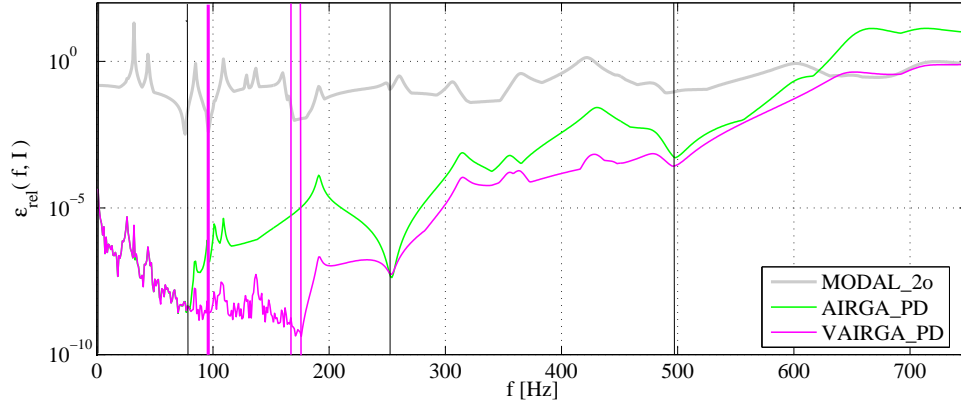
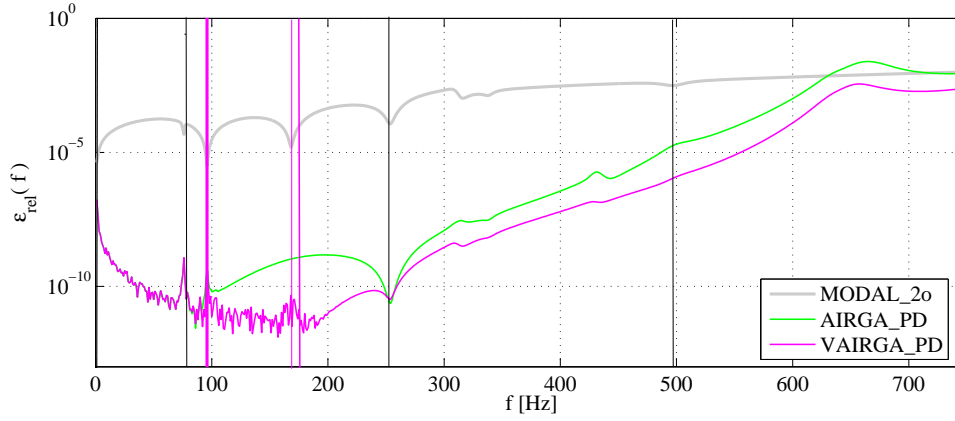
(a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$ (b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung 10.14: Relative Fehler der reduzierten Systeme MODAL_2o, AIRGA_PD und VAIRGA_PD mit verwendeten Entwicklungsstellen in der letzten Iteration der Krylov-Unterraum-Verfahren (magenta: Entwicklungsstellen des Verfahrens VAIRGA_PD, schwarz: Entwicklungsstellen beider Verfahren) - erstes Beispielmodell

reduzierte System AIRGA_ND. Das reduzierte System VAIRGA_2o erzielt auch im Vergleich mit den reduzierten Systemen MIRA_PD und IRGA_PD die besten Approximationsergebnisse bezüglich der hier betrachteten Fehlermaße.

Zweites Beispielmodell

Die Beurteilung der Approximationsergebnisse der Krylov-Unterraum reduzierten Systeme des zweiten Beispielmodells der Dimension 54885, das vier Eingänge und vier Ausgänge besitzt, erfolgt anhand reduzierter Systeme der Dimension 48, die einem modal reduzierten Modell der gleichen Dimension (MODAL_2o) gegenübergestellt und verglichen werden, welches mit dem FEM-Programm MSC.Nastran[®] generiert wird. Die Dimension des modal reduzierten Systems entspricht dabei der Anzahl der Eigenmoden im Intervall [0 Hz, 1264.5 Hz].

In Anhang A.2 sind die Approximationsfehler des mit den verschiedenen Verfahren reduzier-

ten zweiten Beispielmodells der Dimension 32, 40, 48 und 56 dargestellt. Zur ausführlichen Darstellung der Approximationsergebnisse wird die reduzierte Dimension 48 gewählt, da diese Dimension mit den Dimensionen reduzierter Systeme, die aus der Anwendung etablierter Reduktionsverfahren auf Modelle dieser Größenordnung resultieren, vergleichbar ist und die reduzierten Systeme mit dieser Dimension im gewählten Frequenzbereich bereits gute Approximationsergebnisse liefern.

Die Algorithmen der Reduktionsverfahren (Algorithmus 19, Algorithmus 21 und Algorithmus 22) werden dazu mit den Anfangsentwicklungsstellen $\iota \cdot 2\pi$, $100\iota \cdot 2\pi$, $200\iota \cdot 2\pi$, $300\iota \cdot 2\pi$, $400\iota \cdot 2\pi$ und $500\iota \cdot 2\pi$ sowie $\tau = 10^{-30}$, $tol = 10^{-5}$, $f_{max} = 500$ und $\Delta = (500/5) \cdot 2\pi$ (AIRGA_2o) bzw. $\Delta = 0$ (VAIRGA_2o) verwendet. Für die Algorithmen des Verfahrens MIRA_2o (Algorithmus 19) und des Verfahrens IRGA_2o (Algorithmus 21) werden für die Anzahl der Entwicklungen pro Entwicklungsstelle $\hat{k} = 2$ verwendet. Die Dimension der reduzierten Systeme $r = 48$ ergibt sich durch die Anzahl der Anfangsentwicklungsstellen \hat{i} , der Stufe des Krylov-Unterraumes k_i an jeder Entwicklungsstelle s_i und der Anzahl der (linear unabhängigen) Eingänge \tilde{m} damit zu $r = \tilde{m} \cdot \hat{i} \cdot \hat{k} = 6 \cdot 4 \cdot 2 = 48$.

Die Ergebnisse der verschiedenen Reduktionsverfahren sind in der Tabelle 10.6 und die verwendeten Entwicklungsstellen in der letzten Iteration der Verfahren und deren Verwendungsreihenfolge sind in der Tabelle 10.7 bzw. 10.8 zusammengefasst. Da die Berechnung des modal reduzierten Systems mit des FEM-Programms MSC.Nastran[®] erfolgt, die bezüglich der Berechnungszeit und des Speicherbedarfs optimale Algorithmen verwendet, wird hier auf den Vergleich der Berechnungszeit des modal reduzierten Systems mit den übrigen reduzierten Systemen verzichtet. Die Abbildungen 10.15 und 10.17 geben den Verlauf der relativen Fehler (10.3) und (10.5) der verschiedenen reduzierten Modelle wieder. In den Abbildungen 10.16 und 10.18 ist ein Vergleich der Verfahren AIRGA_ND und VAIRGA_ND bzw. AIRGA_PD und VAIRGA_PD dargestellt.

Tabelle 10.6: Ergebnisse der untersuchten Reduktionsverfahren - zweites Beispielmodell

Verfahren	Moden	Zeit [s]	Anzahl Iterationen	Fehler $\epsilon_{max}(\mathcal{I})$ (10.4)	Fehler ϵ_{max} (10.6)
MODAL_2o	48	–	–	39.74	0.82
Verfahren mit Anwendung auf das ungedämpftes System					
MIRA_ND	–	251.85	2	$19.30 \cdot 10^{-6}$	$1.14 \cdot 10^{-6}$
IRGA_ND	–	343.67	2	$1.18 \cdot 10^{-6}$	$61.69 \cdot 10^{-9}$
AIRGA_ND	–	786.98	2	$13.36 \cdot 10^{-3}$	$28.77 \cdot 10^{-6}$
VAIRGA_ND	–	1349.10	2	$14.22 \cdot 10^{-6}$	$0.52 \cdot 10^{-6}$
Verfahren mit Anwendung auf das proportional gedämpfte System					
MIRA_PD	–	337.05	2	$22.10 \cdot 10^{-6}$	$1.20 \cdot 10^{-6}$
IRGA_PD	–	455.61	2	$1.05 \cdot 10^{-6}$	$57.98 \cdot 10^{-9}$
AIRGA_PD	–	1065.90	2	$1.44 \cdot 10^{-3}$	$2.31 \cdot 10^{-6}$
VAIRGA_PD	–	1629.90	2	$8.30 \cdot 10^{-6}$	$0.41 \cdot 10^{-6}$

Tabelle 10.7: Entwicklungsstellen der letzten Iteration der untersuchten Reduktionsverfahren - zweites Beispielmodell

Nr.	Entwicklungsstelle	Nr.	Entwicklungsstelle	Nr.	Entwicklungsstelle
1	$4.92 \cdot 10^{-3} + 48.07 \iota$	2	$8.83 \cdot 10^{-3} + 64.62 \iota$	3	$2.31 \cdot 10^{-2} + 104.74 \iota$
4	$2.78 \cdot 10^{-2} + 115 \iota$	5	$5.30 \cdot 10^{-2} + 158.91 \iota$	6	$7.83 \cdot 10^{-2} + 193.26 \iota$
7	$8.26 \cdot 10^{-2} + 198.5 \iota$	8	$0.18 + 290.00 \iota$	9	$0.22 + 322.24 \iota$
10	$0.31 + 386.44 \iota$	11	$0.34 + 402.57 \iota$	12	$0.38 + 428.30 \iota$
13	$0.44 + 459.32 \iota$	14	$0.52 + 497.59 \iota$		

Tabelle 10.8: Reihenfolge der Verwendung der Entwicklungsstellen in der letzten Iteration der untersuchten Reduktionsverfahren - zweites Beispielmodell

Verfahren	Reihenfolge der verwendeten Entwicklungsstellen
MIRA_ND, MIRA_PD IRGA_ND, IRGA_PD	1, 1, 14, 14, 2, 2, 6, 6, 9, 9, 12, 12
AIRGA_ND	1, 14, 2, 6, 9, 12, 12, 12, 1, 2, 9, 6
AIRGA_PD	1, 14, 2, 6, 9, 12, 12, 12, 1, 2, 9, 12
VAIRGA_ND, VAIRGA_PD	1, 14, 2, 3, 4, 5, 6, 7, 8, 10, 11, 13

Beim Vergleich der Fehlerverläufe in Abbildung 10.15 und Abbildung 10.17 ist ersichtlich, dass Krylov-Unterraum-Verfahren auch bei der Reduktion von großen und umfangreichen Modellen verwendet werden können. Mit allen hier vorgestellten Krylov-Unterraum-Verfahren konnten gute Approximationen des Frequenzganges im gewählten Frequenzintervall erzielt werden.

Die Approximationsergebnisse der Verfahren sind im Vergleich zu dem modal reduzierten Modell der gleichen Dimension durchgehend deutlich besser.

Auch bei diesem Beispielmodell wird die innerhalb eines vorgegebenen Frequenzintervalls erzielbare hohe Genauigkeit der reduzierten Modelle, die mit den in dieser Arbeit vorgeschlagenen Krylov-Unterraum-Verfahren reduziert wurden, deutlich.

Bei diesem Modellbeispiel sind die erzielten Approximationsergebnisse der mit den entwickelten Verfahren bei Anwendung auf ein ungedämpftes System und anschließender proportionaler Dämpfung oder bei direkter Anwendung auf ein proportional gedämpftes System, beinahe identisch. Alle reduzierten Modelle, die aus der Anwendung der Krylov-Unterraum-Verfahren resultieren, besitzen dabei maximale relative Fehler $\epsilon(f, \mathcal{I})$ und $\epsilon(f)$, die im gewählten Frequenzintervall kleiner als $2.2 \cdot 10^{-5}$ sind.

Im Vergleich mit den Ergebnissen des ersten Beispielmodells zeigen sich bessere Approximationen des Originalsystems durch die reduzierten Systeme aller Verfahren. Hierbei ist zu beachten, dass bei der Anwendung der Reduktionsverfahren auf das System des zweiten Beispielmodells

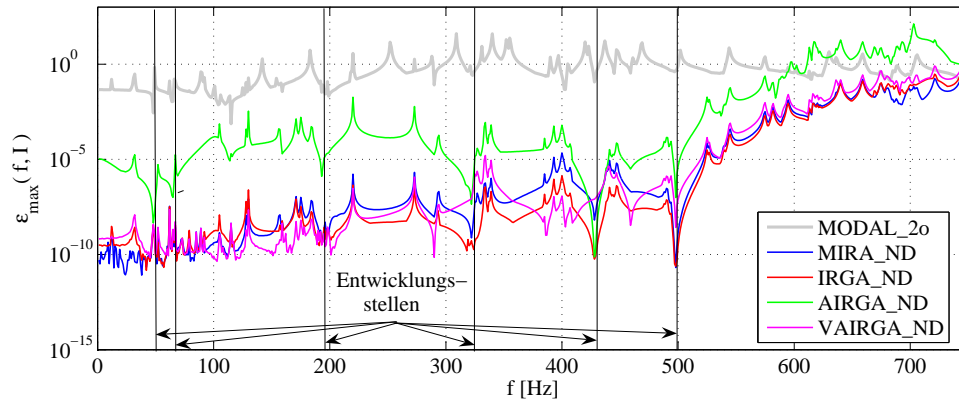
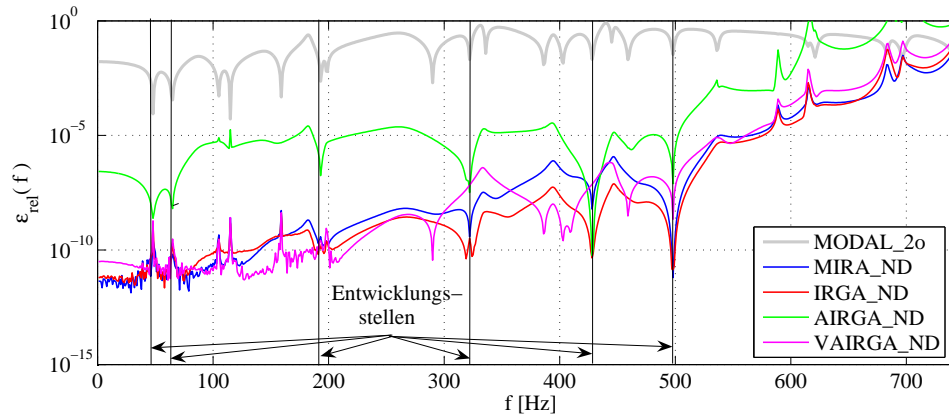

 (a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$

 (b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung 10.15: Relative Approximationsfehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND - zweites Beispielmodell

sechs statt vier Anfangsentwicklungsstellen verwendet werden, was zu reduzierten Systemen mit einer höheren Dimension als bei der Anwendung der Verfahren auf das Systems des ersten Beispielmodells führt und dadurch eine bessere Approximation der Übertragungsfunktion des Originalsystems erwarten lässt.

10.4.5 Zusammenfassung der Resultate

Zusammenfassend lässt sich feststellen, dass mit dem Verfahren AIRGA_2o ein Verfahren basierend auf dem globalen Arnoldi-Algorithmus zur Reduktion von Systemen zweiter Ordnung vorgestellt wird, das die Entwicklungsstellen iterativ und die Stufen der dazu korrespondierenden Krylov-Unterräume adaptiv bestimmt. Werden zudem auf dem Intervall $2\pi i \cdot [1, f_{max}]$ äquidistant verteilte Anfangsentwicklungsstellen verwendet, werden als Eingabeparameter lediglich die Dimension des reduzierten Systems r , die Toleranz tol , ein Wert für die Distanz zweier Entwicklungsstellen und die maximal relevante Frequenz f_{max} als Informationen vom Anwender benötigt.

In der Version VAIRGA_2o des Verfahrens ist der Anwender zudem von der Eingabe einer Di-

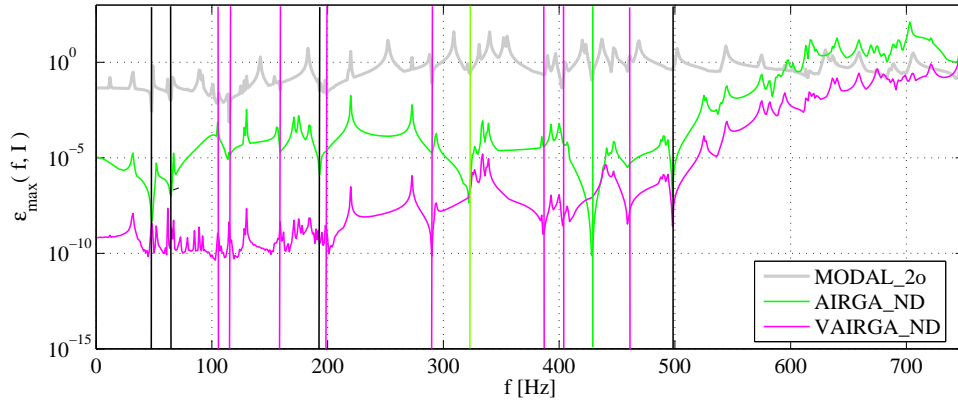
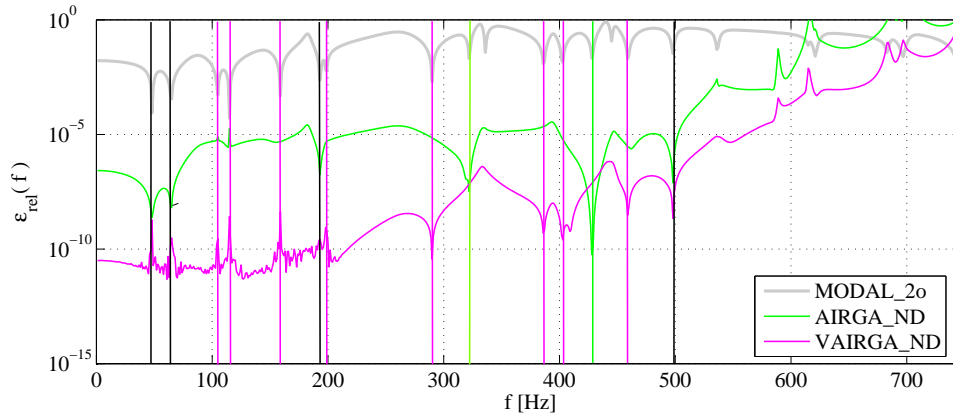
(a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$ (b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung 10.16: Relative Fehler der reduzierten Systeme MODAL_2o, AIRGA_ND und VAIRGA_ND mit verwendeten Entwicklungsstellen in der letzten Iteration der Krylov-Unterraum-Verfahren (grün: Entwicklungsstelle des Verfahrens AIRGA_ND, magenta: Entwicklungsstellen des Verfahrens VAIRGA_ND, schwarz: Entwicklungsstellen beider Verfahren) - zweites Beispielmodell

stanz zweier Entwicklungsstellen enthoben, womit der Anwender dieses Verfahrens weder Kenntnisse noch Erfahrungen bezüglich der Wahl der Entwicklungsstellen und der Stufen der dazu korrespondierenden Krylov-Unterräume benötigt. Damit bildet dieses Verfahren die notwendige Basis für ein weiterführendes automatisiertes Verfahren zur Reduktion von Systemen zweiter Ordnung.

Die aus der Anwendung der Krylov-Unterraum-Verfahren resultierenden reduzierten Systeme zeigen gute Approximation des unteren Frequenzbereiches, dessen Betrachtung bezüglich einer Anwendung der reduzierten Modelle in der Simulation von Werkzeugmaschinen von besonderer Relevanz ist.

Im Vergleich mit den in der Strukturmechanik etablierten Verfahren basierend auf der Methode der modalen Reduktion wird durch die guten Approximationsergebnisse anhand der beiden Beispielmodelle deutlich, dass sich die Krylov-Unterraum-Verfahren als alternative Verfahren

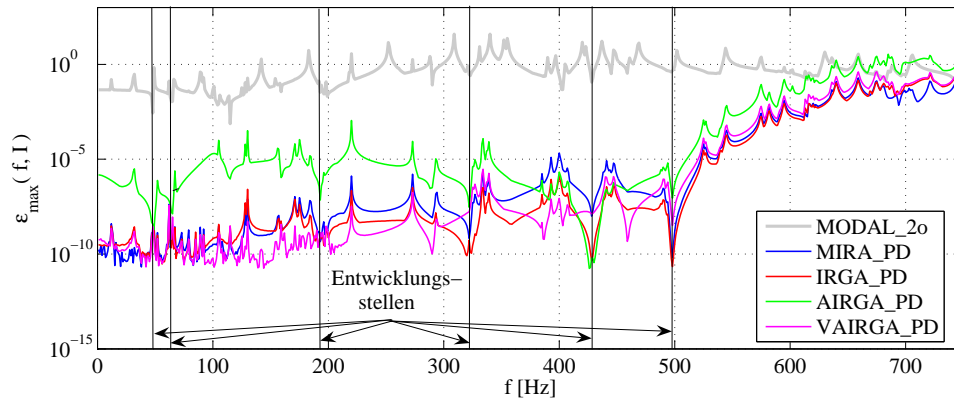
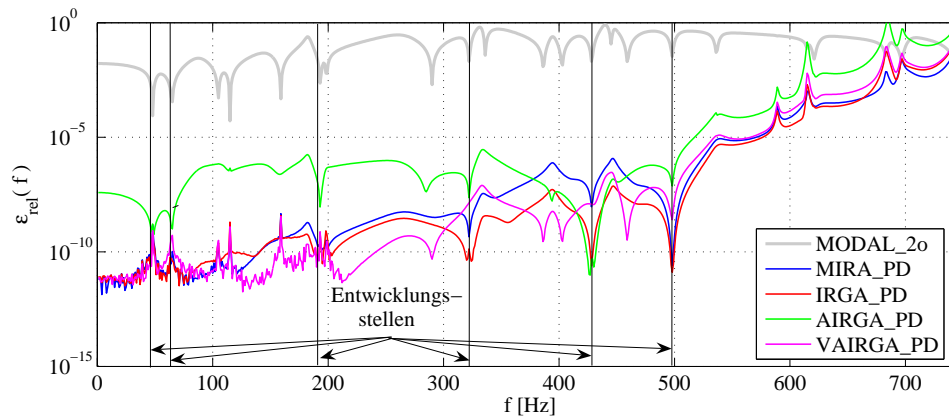

 (a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$

 (b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung 10.17: Relative Approximationsfehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD - zweites Beispielmodell

zur Reduktion von Strukturmodellen gut eignen.

Bei dem Vergleich der Ergebnisse der vier in dieser Arbeit vorgestellten Krylov-Unterraum-Verfahren zeigt sich ein besonderer Vorteil des Verfahrens VAIRGA_2o in Hinblick auf eine Automatisierung des Reduktionsprozesses. Dieser Vorteil liegt in der Verwendung aller potenziellen Entwicklungsstellen auf Grundlage aller Eigenwerte des in der vorhergehenden Reduktions-Iteration berechneten reduzierten Systems im relevanten Frequenzbereich in Kombination mit der adaptiven Bestimmung der Stufen der dazu korrespondierenden Krylov-Unterräumen.

Die Kombination der iterativen Bestimmung der potenziellen Entwicklungsstellen mit Hilfe der im relevanten Frequenzbereich liegenden Eigenwerte des zuvor reduzierten Systems und der Auswahl der Entwicklungsstellen zur sukzessiven Berechnung der Projektionsmatrix \mathbf{V} mit Hilfe der Momentenfehler an den potenziellen Entwicklungsstellen führt zu einer Reduktion der Momentenfehler an den Polstellen der Übertragungsfunktion im betrachteten Frequenzintervall.

Allerdings steigt der Approximationsfehler der mit diesem Verfahren reduzierten Systeme bei höheren Frequenzen schnell an, wenn der niederfrequente Bereich schwer zu approximieren ist. Die Ursache dafür liegt in der Konzentration der bestimmten Entwicklungsstellen in diesem

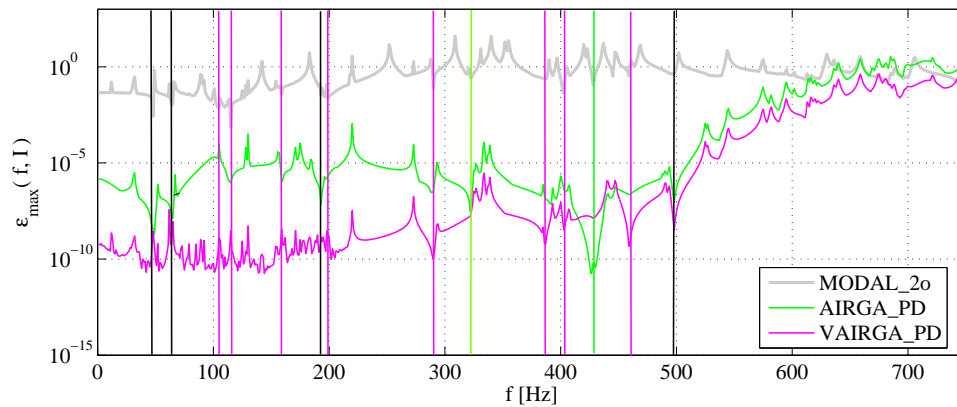
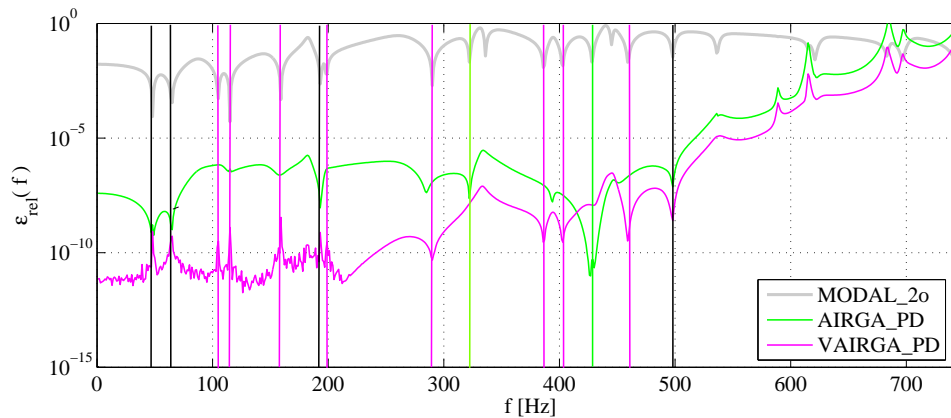
(a) Relativer Fehler $\epsilon_{\max}(f, \mathcal{I})$ (b) Relativer Fehler $\epsilon_{\text{rel}}(f)$

Abbildung 10.18: Relative Fehler der reduzierten Systeme MODAL_2o, AIRGA_PD und VAIRGA_PD mit verwendeten Entwicklungsstellen in der letzten Iteration der Krylov-Unterraum-Verfahren (grün: Entwicklungsstelle des Verfahrens AIRGA_PD, magenta: Entwicklungsstellen des Verfahrens VAIRGA_PD, schwarz: Entwicklungsstellen beider Verfahren) - zweites Beispielmodell

Bereich. Zudem führt die höhere Anzahl von potenziellen Entwicklungsstellen bei der adaptiven Bestimmung neuer Entwicklungsstellen zu einem höheren Aufwand, wodurch das Verfahren deutlich mehr Berechnungszeit als die übrigen Verfahren mit einer beschränkten Anzahl an potenziellen Entwicklungsstellen benötigt.

10.5 Automatische Modellreduktion

In den vorhergehenden Abschnitten konnte gezeigt werden, dass ein Verfahren mit iterativer Bestimmung der potenziellen Entwicklungsstellen und adaptiver Bestimmung der Stufen der verwendeten Krylov-Unterräume zur Reduktion von strukturellen Modellen verwendet werden kann.

Dem Anwender dieses Verfahrens stehen in der Regel jedoch a priori keine Informationen zur optimalen Wahl der Dimension bei einer vorgegebenen Genauigkeit des reduzierten Systems zur Verfügung. In der Praxis ist der Anwender deshalb gezwungen, die Dimension des reduzierten Systems durch die Bestimmung des Approximationsfehlers des reduzierten Systems sukzessive anzupassen, bis eine geforderte Fehlergrenze im gewünschten Frequenzbereich unterschritten wird. Ein Ausweg, diese Prozedur zu automatisieren, besteht in der Verwendung eines geeigneten Fehlerschätzers, der den Approximationsfehler des reduzierten Systems während des Reduktionsprozesses abschätzt und bei Unterschreiten einer vorgegebenen Fehlerschranke den Reduktionsprozess automatisch abbricht. Diese automatische Prozedur ist in Abbildung 10.19 schematisch dargestellt. In diesem Abschnitt wird, aufbauend auf den vorhergehenden Abschnitten, ein neues

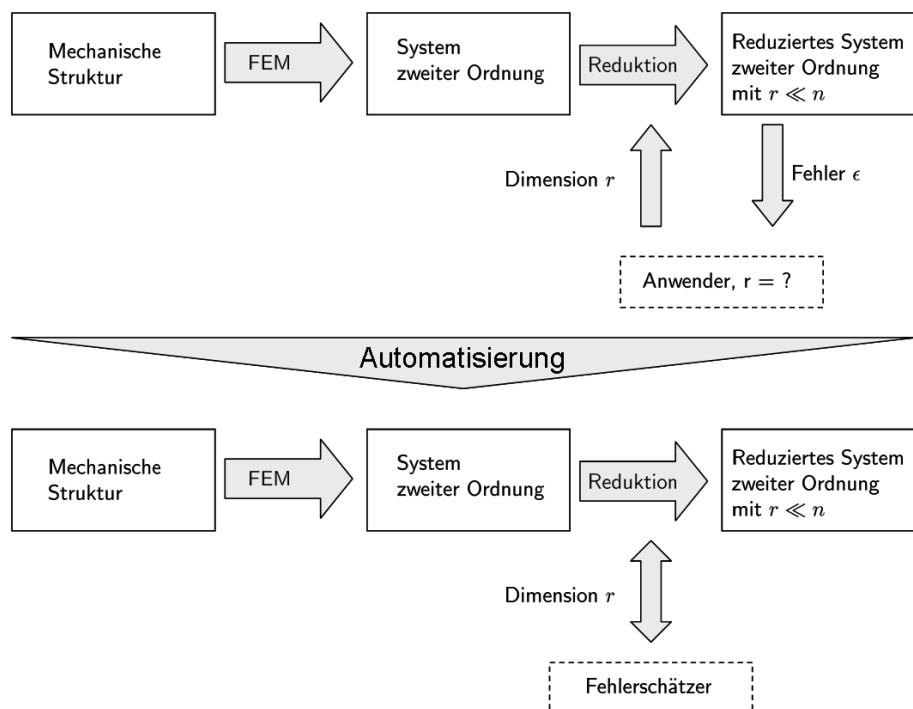


Abbildung 10.19: Schema einer automatischen Reduktion mit Fehlerschätzer - Der Anwender wird der sukzessiven Anpassung der Dimension des reduzierten Systems enthoben

Verfahren zur automatischen Reduktion von Strukturmodellen vorgestellt.

Die grundlegende Bedingung zur Automatisierung des Reduktionsprozesses ist die Existenz eines effizient berechenbaren Fehlerschätzers. In Abschnitt 10.5.1 wird deshalb auf Grundlage des Verfahrens VAIRGA_2o zuvor eine effiziente Methode vorgestellt, um den Fehler eines reduzierten Systems abschätzen zu können. Diese Abschätzung basiert auf der Auswertung der Übertragungsfunktion des originalen und des reduzierten Systems an den potenziellen Entwick-

lungsstellen.

Die vorgeschlagene Methode zur Fehlerschätzung kann verwendet werden, um den Reduktionsprozess so zu automatisieren, dass sich die vom Anwender verlangten Eingaben auf ein Frequenzintervall, auf dem das Originalsystem durch das reduzierte System gut approximiert werden soll, der Anzahl der Anfangsentwicklungsstellen sowie einer maximalen Dimension und einem maximalen Approximationsfehler des reduzierten Systems beschränken.

In der Literatur existieren weitere Methoden der Fehlerschätzung reduzierter Systeme, die anstatt der hier vorgeschlagenen Methode zur Automatisierung des Reduktionsprozesses herangezogen werden können. Die wichtigsten dieser Methoden wurden bereits in Kapitel 9 vorgestellt. In dieser Arbeit wird die oben beschriebene Methode der Fehlerschätzung verwendet, weil sich diese in Kombination mit der iterativen Bestimmung der potenziellen Entwicklungsstellen mit dem geringsten zusätzlichen numerischen Aufwand realisieren lässt.

10.5.1 Abschätzung des Approximationsfehlers

Als Ausgangspunkt bei der Entwicklung des hier vorgeschlagenen Fehlerschätzers dient die Beobachtung, dass die größten Approximationsfehler des reduzierten Systems in der Umgebung der Pole der Übertragungsfunktion des Originalsystems liegen. Da diese Pole Eigenwerte des Originalsystems darstellen und die iterativ bestimmten Entwicklungsstellen der Reduktionsverfahren gegen diese Eigenwerte des Originalsystems konvergieren, wird in deren Umgebung eine gute Approximation der Übertragungsfunktion erzielt.

Auf dieser Erkenntnis aufbauend kann die Betrachtung der Fehler an den Entwicklungsstellen während einer Reduktions-Iteration des Verfahrens VAIRGA_2o verwendet werden, um den Fehler des reduzierten Systems abzuschätzen. Diese Methode der Fehlerschätzung besitzt den Vorteil, dass sie effizient zu berechnen ist. Ein weiterer Vorteil dieser Methode zur Fehlerschätzung liegt in der Möglichkeit verschiedene Normen zur Angabe des lokalen Fehlers zu verwenden. Als geeignete Normen für den lokalen Fehler lassen sich z.B. die in Abschnitt 10.1.1 betrachteten Normen verwenden.

Im Ablauf des Verfahrens VAIRGA_2o (Algorithmus 22) wird in Zeile 11 zu jeder Entwicklungsstelle s_i , der im betrachteten Frequenzbereich liegt, das Residuum

$$\mathbf{R}^{(0)}(s_i) = (s_i^2 \mathbf{M} + s_i \mathbf{D} + \mathbf{K})^{-1} \mathbf{F} \quad (10.59)$$

berechnet. Mit Hilfe der bereits berechneten Residuen (10.59) kann deshalb die Übertragungsfunktion an den Entwicklungsstellen s_i des unreduzierten Systems $\mathbf{H}(s_i)$ durch

$$\mathbf{H}(s_i) = (\mathbf{C}_p + s_i \mathbf{C}_v) \mathbf{R}^{(0)}(s_i) \quad (10.60)$$

effizient berechnet werden. Die für die Berechnung des Fehlers nötige Übertragungsfunktion an der Entwicklungsstelle s_i des reduzierten Systems ist ebenfalls effizient durchführbar, da hierzu nur ein Gleichungssystem der reduzierten Dimension gelöst werden muss. Dazu wird das reduzierte System $\hat{\Sigma}_{2o}$ durch die Projektion (5.17) ermittelt und die Übertragungsfunktion des reduzierten Systems $\hat{\mathbf{H}}(s_i)$ an der Entwicklungsstelle s_i berechnet. Als Fehlerabschätzung kann dann beispielsweise das Fehlermaß (10.4) oder (10.6) verwendet werden.

10.5.2 Automatische Reduktionsverfahren

Mit Hilfe der im vorhergehenden Abschnitt vorgeschlagenen Fehlerabschätzung werden in diesem Abschnitt Verfahren vorgeschlagen, die den Reduktionsprozess automatisch durchführen.

Als Basis der automatisierten Verfahren dient das Verfahren VAIRGA_2o, das durch die Verwendung der oben beschriebenen Fehlerbestimmung automatisch abbricht, wenn eine vorgegebene Fehlerschranke unterschritten wird. Wird in der j -ten Iteration die vorgegebene maximale Dimension des reduzierten Systems erreicht und die Fehlerschranke nicht unterschritten, erfolgt eine weitere Iteration des Verfahrens mit neu bestimmten potenziellen Entwicklungsstellen. Als Abbruchkriterium des Verfahrens dient wie beim Verfahren VAIRGA_2o die \mathcal{H}_2 -Norm der Differenz zweier aufeinander folgender reduzierter Systeme. Wird diese Differenz unterschritten, führt dies zum Abbruch des Verfahrens und zur Ausgabe des maximalen Approximationsfehlers des reduzierten Systems.

Um die Fehlerabschätzung und die Approximation in einem weiteren Frequenzbereich zu erhöhen, werden die automatisierten Verfahren auf Basis des Verfahrens VAIRGA_2o entwickelt. Bei den automatisierten Verfahren werden somit alle potenziellen Entwicklungsstellen verwendet, die durch die in Abschnitt 10.3.2.1 beschriebene Methode der Stellenauswahl ermittelt werden. Um das Verfahren VAIRGA_2o zu automatisieren wird der Algorithmus des Verfahrens (Algorithmus 22) wie folgt modifiziert:

1. Die Parameter für die Deflationstoleranz und die Toleranz werden mit den Werten $\tau = 10-30$ bzw. $tol = 10^{-5}$ festgelegt.
2. Für die Anzahl der zu verwendenden Anfangsentwicklungsstellen wird der Eingangsparameter \hat{i} eingeführt.
3. Die \hat{i} Anfangsentwicklungsstellen werden äquidistant auf dem Intervall $2\pi t \cdot [1, f_{max}]$ bestimmt.
4. In Zeile 11 wird neben der Berechnung des Wertes von $\mathbf{R}^{(0)}(s_i)$ zusätzlich mit Hilfe der Gleichung (10.60) der Wert der Übertragungsfunktion des unreduzierten Systems an der Stelle s_i berechnet und gespeichert.
5. Die **for**-Schleife in Zeile 13 wird durch eine **while**-Schleife ersetzt, die sicher stellt, dass die Matrix \mathbf{V} solange sukzessive um Submatrizen \mathbf{V}_j erweitert wird, bis eine vorgegebene maximale Dimension überschritten und eine Fehlerschranke unterschritten ist.
6. Nach jeder Iteration der **while**-Schleife in Zeile 13 wird der (exakte) Fehler des reduzierten Systems berechnet. Dazu wird in der j -ten Iteration der Schleife das mit der Matrix $\mathbf{V} = [\mathbf{V}_1 \cdots \mathbf{V}_j]$ reduzierte System ermittelt und der Approximationsfehler mit den gespeicherten Daten der Übertragungsfunktion des unreduzierten Systems bezüglich eines geeigneten Fehlermaßes berechnet.

Mit diesen Modifikationen ergibt sich das automatisierte Reduktionsverfahren VAIRGA_2o, dessen Ablauf in Abbildung 10.20 als Ablaufdiagramm dargestellt ist.

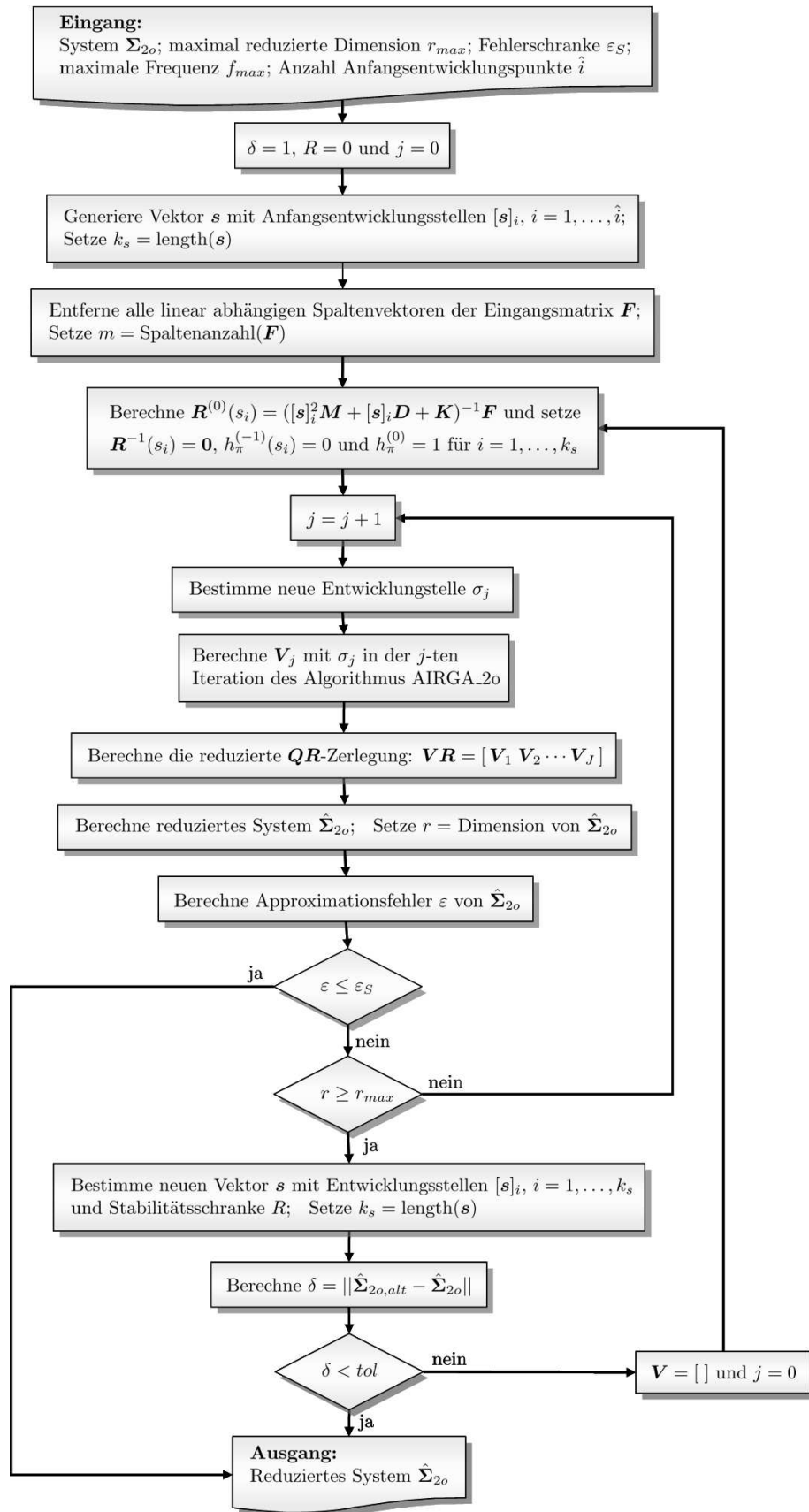


Abbildung 10.20: Ablaufdiagramm des automatisierten Verfahrens VAIRGA_2o (VAIRGA_MM)

10.5.2.1 VAIRGA_MM

Als Basis des hier vorgestellten automatisierten Verfahrens dient das Verfahren VAIRGA_2o mit der in Abschnitt 10.4.4.2 beschriebenen iterativen Entwicklungsstellenwahl. Das durch die Verwendung der oben beschriebenen Fehlerschätzung automatisierte Verfahren bestimmt die Entwicklungsstelle σ_j in der j -ten Iteration des Verfahrens durch Berechnung der j_i -ten Momentenfehler $\epsilon_{j_i}(s_i)$ an den potenziellen Entwicklungsstellen s_i , $i = 1, \dots, k_s$ mit (10.55) oder des maximalen Momentenfehlers mit der Gleichung (10.57), wohingegen die maximalen relativen Fehler $\epsilon_{max}(\mathcal{I})$ (10.4) oder ϵ_{max} (10.6) als Approximationsfehlerschätzer des reduzierten Systems dienen.

Da eine korrekte Fehlerabschätzung des reduzierten Systems nur mit Kenntnis der verwendeten Dämpfung des Systems berechnet werden kann, wird das Verfahren für die Anwendung auf proportional gedämpfte Originalsysteme zweiter Ordnung der Form (3.1) mit $\mathbf{D} \neq \mathbf{0}$ entwickelt. Das mit Hilfe der oben beschriebenen Modifikationen erzielte automatisierte Reduktionsverfahren wird hier VAIRGA_MM genannt, wobei der Anhang _MM für maximaler Momentenfehler steht. Das Verfahren, das den Fehler $\epsilon_{max}(\mathcal{I})$ als Fehlerschätzer des reduzierten Systems verwendet, wird VAIRGA_MM_MT und das Verfahren, das den Fehler ϵ_{max} als Fehlerschätzer des reduzierten Systems verwendet, wird hier VAIRGA_MM_MF genannt.

Als Eingabeparameter der automatisierten Verfahren werden nur die Vorgabe des zu approximierenden Frequenzbereichs, die Anzahl der Anfangsentwicklungsstellen und die maximale Dimension des reduzierten Modells benötigt. Der Anwender erhält als Ausgabeinformation den geschätzten maximalen Approximationsfehler des reduzierten Systems. Bei zusätzlicher Eingabe einer Fehlerschranke berechnet das Verfahren ein reduziertes Modell das einen geschätzten Approximationsfehler besitzt der unter der angegebenen Fehlerschranke liegt. Ist eine Reduktion mit der vorgegebenen reduzierten Dimension und des vorgegebenen Approximationsfehlers nicht möglich, wird ein reduziertes Modell mit maximaler reduzierter Dimension berechnet und dessen geschätzter maximaler relativer Approximationsfehler ausgegeben.

10.5.2.2 Weitere automatische Reduktionsverfahren

Wie bei dem zuvor dargestellten automatisierten Reduktionsverfahren VAIRGA_MM dient das Verfahren VAIRGA_2o als Basis zweier weiterer automatisierter Verfahren, die hier vorgestellt und analysiert werden sollen. Diese Verfahren unterscheiden sich gegenüber dem Verfahren VAIRGA_MM in der Auswahl neuer Entwicklungsstellen. Anstatt die in Abschnitt 10.4.4.2 beschriebene Entwicklungsstellenwahl mit Hilfe des Momentenfehlers an den potenziellen Entwicklungsstellen s_i als Auswahlkriterium weiter zu verfolgen, dient bei diesen Verfahren der relative Approximationsfehler $\epsilon_{max}(s_i, \mathcal{I})$ (10.3) oder $\epsilon_{max}(s_i)$ (10.5) als Kriterium zur Auswahl der Entwicklungsstelle σ_j in der j -ten Iteration der Verfahren mit Hilfe von

$$\sigma_j = \arg \max \{ \epsilon_{max}(s_i, \mathcal{I}) : i = 1, \dots, k_s \} \quad (10.61)$$

bzw.

$$\sigma_j = \arg \max \{ \epsilon_{max}(s_i) : i = 1, \dots, k_s \}. \quad (10.62)$$

Als Fehlerschätzer des reduzierten Systems dient wiederum der relative Approximationsfehler $\epsilon_{max}(\mathcal{I})$ (10.4) oder ϵ_{max} (10.6).

Da die Bestimmung des Approximationsfehlers als Fehlerschätzer in der $(j - 1)$ -ten Iteration bereits berechnet wurde, sind keine weiteren Berechnungen für den Auswahlprozess notwendig. Es ist lediglich notwendig, die Entwicklungsstelle mit dem maximalen relativen Approximationsfehler zu speichern und in der folgenden Iteration als Entwicklungsstelle zu verwenden.

Das Verfahren, das den Fehler $\epsilon_{max}(\mathcal{I})$ als Fehlerschätzer des reduzierten Systems verwendet, wird VAIRGA_MT und das Verfahren, das den Fehler ϵ_{max} als Fehlerschätzer des reduzierten Systems verwendet, wird VAIRGA_MF genannt.

Die Tabelle 10.9 faßt nochmals die verwendeten Fehlermaße, die zur Bestimmung der Entwicklungsstellen und zur Abschätzung des Approximationsfehlers der beschriebenen Verfahren verwendet wurden, zusammen.

Tabelle 10.9: Verwendete Fehlermaße zur Bestimmung der Entwicklungsstellen und zur Abschätzung des Approximationsfehlers des reduzierten Systems der beschriebenen Verfahren

Verfahren	Fehlermaß zur Bestimmung von σ_j	Gleichung des Fehlermaßes	Gleichung zur Bestimmung von σ_j	Fehlermaß zur Abschätzung des Approximationsfehlers	Gleichung des Fehlermaßes
VAIRGA_MT	$\epsilon_{rel}(s_i, \mathcal{I})$	(10.3)	(10.61)	$\epsilon_{max}(\mathcal{I})$	(10.4)
VAIRGA_MM_MT	$\epsilon_{j_i}(s_i)$	(10.55)	(10.57)	$\epsilon_{max}(\mathcal{I})$	(10.4)
VAIRGA_MF	$\epsilon_{rel}(s_i)$	(10.5)	(10.62)	ϵ_{max}	(10.6)
VAIRGA_MM_MF	$\epsilon_{j_i}(s_i)$	(10.55)	(10.57)	ϵ_{max}	(10.6)

10.5.3 Diskussion der automatischen Reduktion

Im folgenden sollen zwei Beispiele für die automatische Durchführung des Reduktionsprozesses angegeben werden. Dazu werden die zwei Beispielm Modelle herangezogen. Das hier betrachtete Frequenzintervall ist $[0 \text{ Hz}, 500 \text{ Hz}]$. Da eine effizient berechnete Fehlerschätzung nur mit vorhergehender Kenntnis der verwendeten proportionaler Dämpfung erzielt wird, werden zur Diskussion des automatischen Verfahrens ausschließlich die Verfahren mit Anwendung auf Systeme betrachtet, die bereits mit $\mathbf{D} \neq \mathbf{0}$ in proportional gedämpfter Form (3.1) vorliegen.

Erstes Beispielm Modell

Zur Validierung der Reduktionsergebnisse wird das erste Beispielm Modell, das die Dimension 4983 sowie vier Eingänge und sechs Ausgänge besitzt, jeweils mit den vier automatisierten Verfahren VAIRGA_MM_MT, VAIRGA_MM_MF, VAIRGA_MT und VAIRGA_MF reduziert. Als Vergleich wird den Approximationsergebnissen der reduzierten Systeme das Approximationsergebnis eines modal reduzierten Systems der Dimension 32 gegenüber gestellt, welches mit dem FEM-Programm MSC.Nastran[®] generiert wurde. Dazu werden die Verfahren mit der

Eingabe einer maximalen reduzierten Dimension $r_{max} = 50$ einer Anzahl der Anfangsentwicklungsstellen $\hat{i} = 4$ und einer maximalen relevanten Frequenz $f_{max} = 500$ verwendet. Da das Originalsystem bezüglich des Fehlermaßes $\epsilon_{max}(\mathcal{I})$ schwer zu approximieren ist, werden die Verfahren VAIRGA_MM_MT und VAIRGA_MT, die dieses Fehlermaß zur Abschätzung des Approximationsfehlers verwenden, mit der Fehlerschranke $\epsilon_S = 10^{-2}$ verwendet. Für die Verfahren VAIRGA_MM_MF und VAIRGA_MF, die das Fehlermaß ϵ_{max} zur Abschätzung des Approximationsfehlers verwenden, wird hingegen eine Fehlerschranke $\epsilon_S = 10^{-4}$ verwendet.

Zur Verifikation des geschätzten Fehlers wird der exakte Fehler im Frequenzintervall $[0 \text{ Hz}, 500 \text{ Hz}]$ bestimmt und mit den geschätzten maximalen Approximationsfehlern verglichen. Die Tabelle 10.10 und Abbildung 10.21 geben die Ergebnisse des Vergleichs wieder. Alle automatischen Verfahren konvergierten nach zwei Iterationen. Die Ergebnisse der verschiedenen Reduktionsverfahren sind in der Tabelle 10.10 zusammengefasst. Die Abbildung 10.21(a) gibt den Verlauf des relativen Fehlers $\epsilon_{rel}(f, \mathcal{I})$ (10.3) der reduzierten Systeme VAIRGA_MT und VAIRGA_MM_MT und die Abbildung 10.21(b) gibt den Verlauf des relativen Fehlers $\epsilon_{rel}(f)$ (10.5) der reduzierten Systeme VAIRGA_MF und VAIRGA_MM_MF wieder.

Tabelle 10.10: Ergebnisse der verwendeten automatischen Verfahren - erstes Beipielmodell

Verfahren	Zeit [s]	reduzierte Dimension	Fehler $\epsilon_{max}(\mathcal{I})$, geschätzt	Fehler $\epsilon_{max}(\mathcal{I})$, exakt	Fehler ϵ_{max} , geschätzt	Fehler ϵ_{max} , exakt
VAIRGA_MT	16.33	24	$16.13 \cdot 10^{-3}$	0.78	—	—
VAIRGA_MM_MT	19.92	28	$3.01 \cdot 10^{-3}$	0.05	—	—
VAIRGA_MF	15.56	24	—	—	$37.51 \cdot 10^{-6}$	$37.51 \cdot 10^{-6}$
VAIRGA_MM_MF	21.61	32	—	—	$65.52 \cdot 10^{-6}$	$7.13 \cdot 10^{-6}$

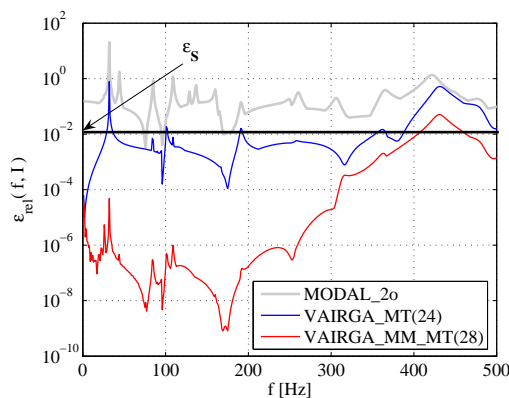
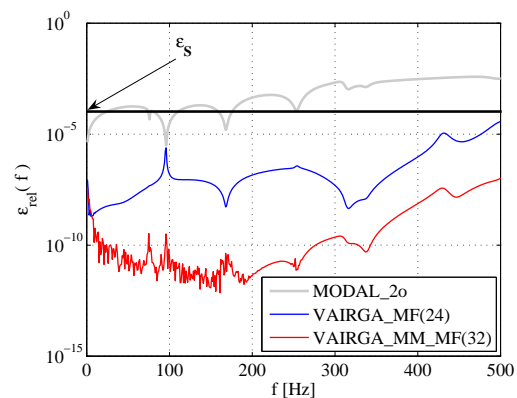

 (a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$

 (b) Relativer Fehler $\epsilon_{rel}(f)$

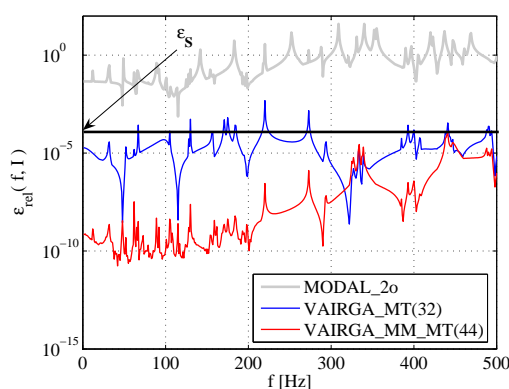
 Abbildung 10.21: Gewählte Fehlerschranken ϵ_S und exakter Approximationsfehler des automatisch reduzierten Systems - erstes Beipielmodell

Zweites Beispielmodell

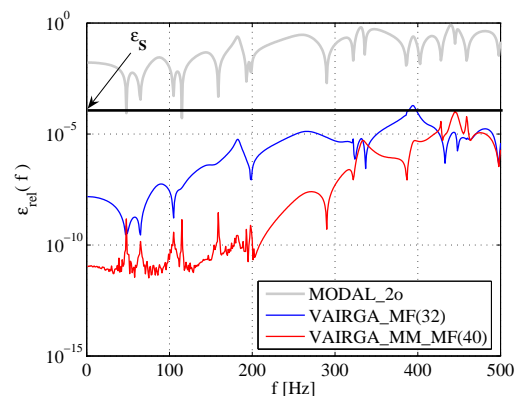
Mit den selben Verfahren wird zusätzlich eine Reduktion des zweiten Modells, das die Dimension 54 885 sowie vier Eingänge und vier Ausgänge besitzt, durchgeführt, wobei als Eingabeparameter eine maximale reduzierte Dimension $r_{max} = 50$, eine Fehlerschranke von $\epsilon_S = 10^{-4}$, eine Anzahl der Anfangsentwicklungsstellen $\hat{i} = 5$ und eine maximale relevanten Frequenz $f_{max} = 500$ gewählt werden. Zur Verifikation des geschätzten Fehlers wird der exakte Fehler im Frequenzintervall $[0 \text{ Hz}, 500 \text{ Hz}]$ bestimmt und mit den geschätzten maximalen Approximationsfehlern verglichen, siehe Tabelle 10.11 und Abbildung 10.22. Als Vergleich wird den Approximationsergebnissen der reduzierten Systeme das Approximationsergebnis eines modal reduzierten Systems der Dimension 48 gegenüber gestellt, welches mit dem FEM-Programm MSC.Nastran® generiert wurde. Alle automatischen Verfahren konvergierten nach zwei Iterationen. Die Ergebnisse der verschiedenen Reduktionsverfahren sind in der Tabelle 10.11 zusammengefasst. Die Abbildung 10.22(a) gibt den Verlauf des relativen Fehlers $\epsilon_{rel}(f, \mathcal{I})$ (10.3) der reduzierten Systeme VAIRGA_MT und VAIRGA_MM_MT und die Abbildung 10.22(b) gibt den Verlauf des relativen Fehlers $\epsilon_{rel}(f)$ (10.5) der reduzierten Systeme VAIRGA_MF und VAIRGA_MM_MF wieder.

Tabelle 10.11: Ergebnisse der verwendeten automatischen Verfahren - zweites Beispielmodell

Verfahren	Zeit [s]	red. Dimen- sion	Fehler $\epsilon_{max}(\mathcal{I})$, geschätzt	Fehler $\epsilon_{max}(\mathcal{I})$, exakt	Fehler ϵ_{max} , geschätzt	Fehler ϵ_{max} , exakt
VAIRGA_MT	837.09	32	$0.15 \cdot 10^{-3}$	$4.82 \cdot 10^{-3}$	—	—
VAIRGA_MM_MT	1333.5	44	$25.24 \cdot 10^{-6}$	$0.18 \cdot 10^{-3}$	—	—
VAIRGA_MF	832.97	32	—	—	$93.63 \cdot 10^{-6}$	$0.19 \cdot 10^{-3}$
VAIRGA_MM_MF	1146.6	40	—	—	$90.31 \cdot 10^{-6}$	$1.16 \cdot 10^{-3}$



(a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$



(b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung 10.22: Gewählte Fehlerschranken ϵ_S und exakter Approximationsfehler des automatisch reduzierten Systems - zweites Beispielmodell

Die Approximationsergebnisse der automatisch reduzierten Modelle machen deutlich, dass die Fehlerbestimmung an den potenziellen Entwicklungsstellen zur Automatisierung des Reduktionsprozesses verwendet werden kann.

Bei der Betrachtung der Approximationsergebnisse der reduzierten Systeme zeigen sich allerdings Unterschiede bei den Fehlerverläufen und den Schätzungen des maximalen Approximationsfehlers der reduzierten Systeme. Die Anwendung der Verfahren VAIRGA_MM_MT und VAIRGA_MM_MF resultiert in reduzierten Systemen mit größerer Dimension und genauerer Approximation des reduzierten Systems als die Anwendung der Verfahren VAIRGA_MF und VAIRGA_MT. Dabei ist eine besonders gute Approximation im unteren Frequenzbereich zu beobachten. Die Ursache dafür liegt in der Auswahl der Entwicklungsstellen über den Momentenfehler bezüglich der F -Norm, was zu einer häufigeren Auswahl von Entwicklungsstellen im unteren Frequenzbereich ohne Verbesserung der Approximation im höheren Frequenzbereich und damit zu einer höheren Dimension der mit diesem Verfahren reduzierten Systeme führt.

10.5.4 Zusammenfassung der Resultate

Bei einem Vergleich der reduzierten Systeme der Verfahren, die den maximalen Fehler der Übertragungsfunktion (VAIRGA_MT, VAIRGA_MM_MT) bzw. den maximalen Fehler bzgl. der F -Norm der Übertragungsfunktion (VAIRGA_MF, VAIRGA_MM_MF) verwenden, ist eine bessere Einhaltung der Fehlerschranke des reduzierten Systems zu beobachten. Die Ursache darin ist in der Verwendung der F -Norm zu sehen, die den Fehler aller Übertragungspfade gemittelt in die Abschätzung des relativen Approximationsfehlers mit einbezieht. Der maximale relative Approximationsfehler aller Übertragungspfade weist eine höhere Volatilität auf, wodurch die Verfahren, die diese Art der Fehlerbestimmung verwenden, eine schlechtere Schätzung des tatsächlichen Fehlers ausgeben. Dies liegt an der nur lokalen Gültigkeit der Fehlerschätzung an den Entwicklungsstellen. Besonders deutlich wird dies anhand von reduzierten Systemen geringer Dimension, die aus der Anwendung des Verfahrens VAIRGA_MT resultieren, siehe Abbildung 10.21(a) bei der Frequenz von 450 Hz.

Nach Betrachtung der Approximationsergebnisse der reduzierten Systeme, die mit den Methoden zur Automatisierung des Verfahrens VAIRGA_2o erzielt wurden, lässt sich zusammenfassend feststellen, dass die Verfahren, die eine Auswahl der Entwicklungsstellen mit Hilfe des Momentenfehlers durchführen zur Automatisierung des Reduktionsprozesses besser geeignet sind als die Verfahren, die eine Auswahl der Entwicklungsstellen mit Hilfe des maximalen Fehler der Übertragungsfunktion durchführen.

Kapitel 11

Modellreduzierte Simulation von Werkzeugmaschinen

Bei den Untersuchungen der Krylov-Unterraum reduzierten Modelle steht neben der Untersuchung der Approximationsergebnisse im Frequenzbereich auch die Untersuchung der Approximationseigenschaften der reduzierten Modelle bei der Simulation von Werkzeugmaschinen im Vordergrund.

Deshalb werden an dieser Stelle die Approximationsergebnisse der reduzierten Modelle der Verfahren MIRKA_2o, IRGA_2o, AIRGA_2o und VAIRGA_2o sowie der automatisierten Verfahren VAIRGA_MM_MT, VAIRGA_MM_MF, VAIRGA_MT und VAIRGA_MF bei der Simulation von Lastfällen und nach Kopplung mit einem Bearbeitungsprozessmodell untersucht und mit den Approximationsergebnissen eines modal reduzierten Systems derselben Dimension verglichen. Da das Originalsystems des zweiten Beispielsmodells eine so hohe Dimension aufweist, dass eine Verifikation der Simulationsergebnisse nicht mehr durchführbar ist, wird als Beispielsmodell zur Untersuchung der Simulationsergebnisse das erste Beispielsmodell der abstrahierten Werkzeugmaschine herangezogen.

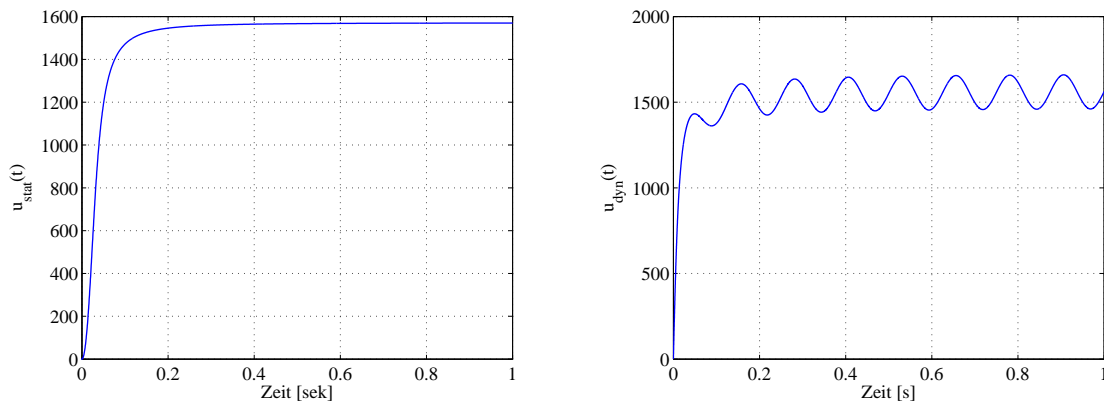
In Abschnitt 11.1 erfolgt die Untersuchung der mit den verschiedenen Krylov-Unterraum-Verfahren reduzierten Modelle erzielten Simulationsergebnisse anhand eines statischen und eines dynamischen Lastfalls. Im darauf folgenden Abschnitt 11.2 erfolgt die Untersuchung der Approximationsergebnisse nach Einbringung der mit den verschiedenen Krylov-Unterraum-Verfahren reduzierten Modelle in die Simulation eines Lageregelkreises. Die Einbindung der reduzierten Modelle erfolgt dabei in die Simulationsumgebung MATLAB/Simulink®. In den jeweiligen Abschnitten werden die Approximationseigenschaften der mit den verschiedenen Krylov-Unterraum-Verfahren reduzierten Modelle mit den Approximationsergebnissen eines modal reduzierten Modells verglichen und diskutiert.

11.1 Untersuchung von Lastfällen

Bei der Simulation und der Reglerauslegung von Werkzeugmaschinen ist die Simulation von schnellen Verfahrbewegungen unter hoher Last, den sogenannten Lastfällen, von besonderem

Interesse und sollten durch ein reduziertes Modell gut approximiert werden. Deshalb sind neben den Untersuchungen der Approximationsergebnisse der reduzierten Modelle im Frequenzbereich auch die Approximationsergebnisse im Zeitbereich von Bedeutung.

Als Beispiele von Lastfällen werden hier jeweils ein statischer und ein dynamischer Lastfall betrachtet. In beiden Fällen handelt es sich um das Aufbringen einer Last auf einen Punkt der mechanischen Struktur, die durch eine (zeitabhängige) Eingangsfunktion $u(t)$ simuliert wird. Bei dem hier betrachteten ersten Beispielmmodell wird das zeitabhängige Verhalten an den sechs Ausgängen der Struktur (y-Rotation Motorwelle, y-Rotation Spindelen- de, y-Translation am Führungsschuh, x-, y-, z-Verlagerung am TCP) nach Aufbringen einer Last untersucht, die durch eine Lastfunktion beschrieben wird, die auf den ersten Eingang (y-Rotation der Motorwelle) wirkt. Dabei wird als statischer Lastfall eine zeitabhängige Last, die durch die Eingangsfunktion $u_{stat}(t) = 1000 \cdot \arctan(1000 \cdot t^2)$ beschrieben wird, verwendet. Beim dynamischen Lastfall wird eine zeitabhängige Last, die durch die Eingangsfunktion $u_{dyn}(t) = 1000 \cdot \arctan(100t) + 100 \cdot \sin(16\pi t)$ beschrieben verwendet. Die Abbildung 11.1 stellt die Lastfunktionen $u_{stat}(t)$ und $u_{dyn}(t)$ graphisch dar.



(a) statische Lastfall: verwendete Eingangsfunktion $u_{stat}(t) = 1000 \cdot \arctan(1000 \cdot t^2)$ (b) Dynamischer Lastfall: verwendete Eingangsfunktion $u_{dyn}(t) = 1000 \arctan(100t) + 100 \sin(16\pi t)$

Abbildung 11.1: Verwendete Funktionen zur Simulation eines statischen bzw. eines dynamischen Lastfalles

Die Bewertung der Approximationsgüte im Zeitbereich ist erheblich schwerer vorzunehmen als im Frequenzbereich, da die Wahl des Gleichungssystemlösers (Solver¹), der zur Bestimmung der Lösung des dem Modell zu Grund liegenden Systems verwendet wird, und die Wahl der verwendeten Parameter des Solvers einen nicht unerheblichen Einfluß auf die Ergebnisse haben können. Um die Approximationsergebnisse der reduzierten Systeme unter den gleichen Bedingungen vergleichen zu können wird deshalb zur Lösung aller Gleichungen des Zeitbereichs der selbe Solver und die selben Zeitpunkte, an denen die Gleichungen ausgewertet werden, verwendet.

Da das Originalsystem als ein System differenzial-algebraischer Gleichungen (DAE) vorliegt, wird zur Berechnung einer Lösung des Gleichungssystems der in MATLAB[®] implementier-

¹Englisch: Löser, dieser Ausdruck ist auch im deutschen Sprachgebrauch als Kurzform für den Ausdruck *Gleichungssysteml ser* gebr uchlich.

te Differenzialgleichungslöser `ode23t` verwendet. Neben dem Differenzialgleichungslöser `ode15s` ermöglicht dieser Solver die Berechnung einer Lösung der auftretenden Gleichungssysteme, ohne die Simulationsumgebung MATLAB/Simulink[®] verlassen zu müssen. Die Implementierung des Solvers `ode23t` basiert auf der Trapezregel und ist zur Lösung von DAE-Systemen mit Index 1 verwendbar, für nähere Informationen zur optimalen Wahl eines geeigneten Solvers zur Lösung von Systemen differenzial-algebraischer Gleichungen sei auf [162] verwiesen.

Zur Untersuchung der Ergebnisse bei Simulationen der oben beschriebenen Lastfälle mit Hilfe reduzierter Modelle, werden die Modelle `MIRA_ND`, `IRGA_ND`, `AIRGA_ND` und `VAIRGA_ND` sowie `MIRA_PD`, `IRGA_PD`, `AIRGA_PD` und `VAIRGA_PD` mit der reduzierten Dimension $r = 32$ verwendet, die aus der Reduktion des ersten Beispielmmodells einer abstrahierten Werkzeugmaschine hervorgehen. Die mit Krylov-Unterraum-Verfahren reduzierten Modelle werden dabei einem modal reduzierten Modell der gleichen Dimension gegenübergestellt und verglichen, welches mit dem FEM-Programm MSC.Nastran[®] generiert wird. Diese Dimension des modal reduzierten Systems entspricht dabei der Anzahl der Eigenmoden im Intervall $[0 \text{ Hz}, 1464.1 \text{ Hz}]$.

Für die Berechnung der reduzierten Systeme werden die Algorithmen der Verfahren `MIRA_2o`, `IRGA_2o`, `AIRGA_2o` und `VAIRGA_2o` (Algorithmus 19, Algorithmus 21 und Algorithmus 22) mit den Anfangsentwicklungsstellen $\iota \cdot 2\pi$, $166\iota \cdot 2\pi$, $332\iota \cdot 2\pi$ und $500\iota \cdot 2\pi$ sowie $\tau = 10^{-30}$, $tol = 10^{-5}$ und $\varepsilon = (500/3) \cdot 2\pi$ (`AIRGA_2o`) bzw. $\Delta = 0$ (`VAIRGA_2o`) verwendet.

Zur Untersuchung der Ergebnisse bei Verwendung von automatisierten Verfahren werden die reduzierten Modelle `VAIRGA_MT`, `VAIRGA_MF`, `VAIRGA_MM_MT` und `VAIRGA_MM_MF` verwendet, die aus der Anwendung der automatischen Reduktionsverfahren resultieren und die reduzierte Dimension $r = 32$ besitzen. Damit alle automatisch reduzierten Systeme die Dimension $r = 32$ aufweisen, wird die Fehlerschranke ϵ_S so klein gewählt, dass die Verfahren bei Erreichen der maximalen reduzierten Dimension, die mit $r_{max} = 32$ gewählt wird, abbrechen.

Bei den Krylov-Unterraum-Verfahren ist zu beachten, dass man komplexwertige Projektionsmatrizen \mathbf{V} erhält, da die Verfahren komplexe Entwicklungsstellen verwenden. Um reellwertige Projektionsmatrizen zu erhalten, werden die komplexwertigen Projektionsmatrizen mit der Methode, die in Abschnitt 8.4.3 bzw. 8.4.4 beschrieben wird, reellwertig gemacht. Dazu erfolgt eine Aufspaltung der Projektionsmatrizen wie in (8.57) bzw. (8.87) und nachfolgend eine Rank-Revealing-**QR**-Zerlegung (8.58) bzw. (8.88), wodurch sich die Spaltenanzahl der Projektionsmatrizen verdoppelt. Durch Anwendung dieser Methode auf die mit den Krylov-Unterraum-Verfahren generierten Projektionsmatrizen \mathbf{V} erhält man somit reduzierte Systeme zweiter Ordnung der Dimension $r = 64$.

Zur Beurteilung des Approximationsfehlers im Zeitbereich wird das Fehlermaß $\delta_{abs}(t, \mathcal{I})$ (10.8) mit $\mathcal{I} = \{1, \dots, q = 6\} \times \{1\}$ verwendet, das bedeutet, dass der maximale Approximationsfehler aller Übertragungspfade vom ersten Eingang zu allen sechs Ausgängen betrachtet wird.

In den Abbildungen 11.2, 11.3 und 11.4 sind die Ergebnisse im Zeitbereich dargestellt. Die Abbildung gibt den Verlauf des absoluten Fehlers $\delta_{abs}(t, \mathcal{I})$ (10.8) der verschiedenen reduzierten Modelle wieder. (Im Falle des dynamischen Lastfalles liegen alle dargestellten Kurven übereinander, so dass diese nicht unterscheidbar sind.)

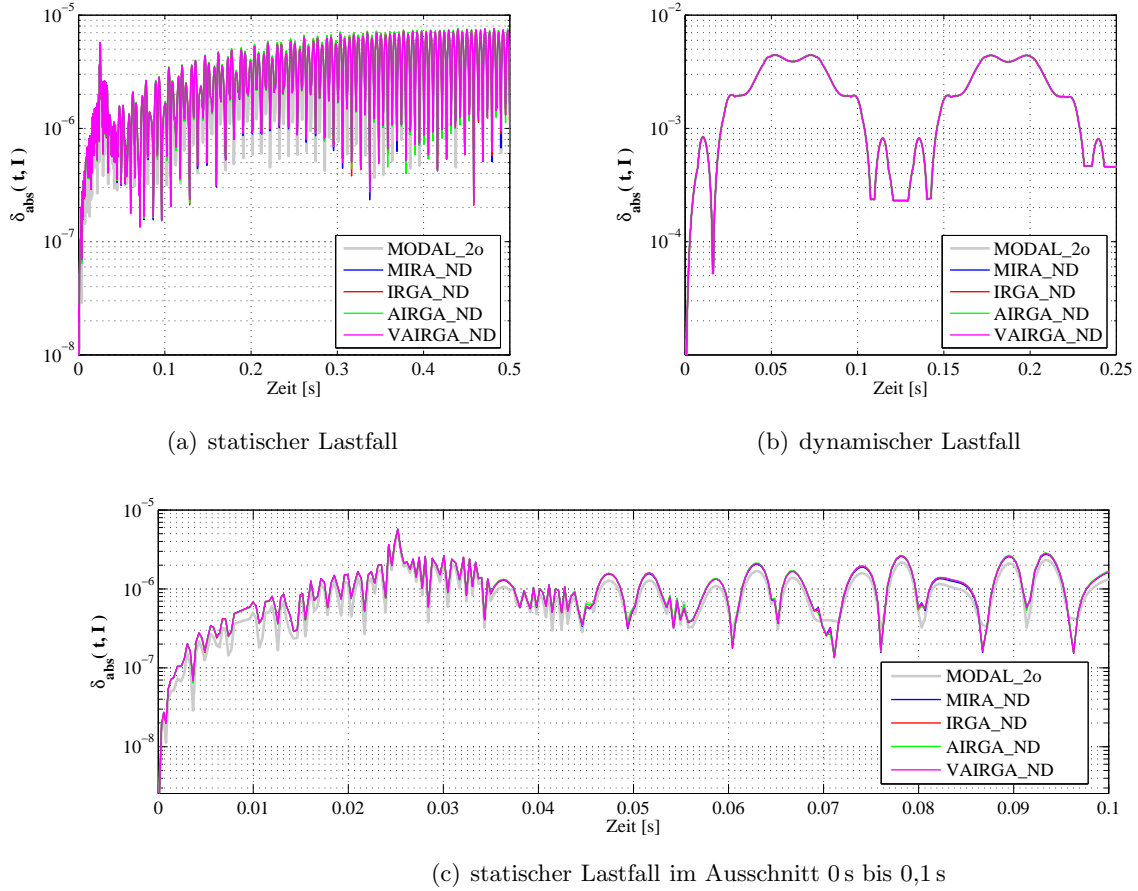


Abbildung 11.2: Absoluter Approximationsfehler $\delta_{abs}(t, \mathcal{I})$ der Simulationsergebnisse für Beispiele eines statischen und eines dynamischen Lastfalles des ersten Beispielmodells, mit den Verfahren angewendet auf das ungedämpfte System.

Resultate

Die durch die in dieser Arbeit entwickelten Verfahren reduzierten Modelle besitzen annähernd den gleichen Approximationsfehler, wie das modal reduzierte System der selben Dimension.

Dabei ist bei den hier gezeigten Ergebnissen zu beachten, dass das modal reduzierte System alle dominanten Moden im betrachteten Intervall $[0 \text{ Hz}, 1464.1 \text{ Hz}]$ enthält. Bei der Verwendung einer Projektionsmatrix, die auf einer zufälligen Auswahl von Moden basiert, kann im allgemeinen die hier getroffene optimale Auswahl der Moden nicht garantiert werden. Somit erzielt man mit dem hier verwendeten modal reduzierten System die bestmögliche Approximation des Zeitbereiches aller modal reduzierten Systeme auf dem betrachteten Zeitintervall.

Unter Beachtung dieser Tatsache sowie unter Beachtung des annähernd identischen Fehlerverlaufs der Krylov-Unterraum reduzierten Systeme mit dem Fehlerverlauf des modal reduzierten Systems, lässt sich feststellen, dass die hier verwendeten Krylov-Unterraum-Verfahren geeignete Verfahren darstellen, um reduzierte Systeme zu realisieren, die zur Simulation von Lastfällen geeignet sind.

Darüber hinaus weisen diese Verfahren gegenüber Verfahren der modalen Reduktion zudem den

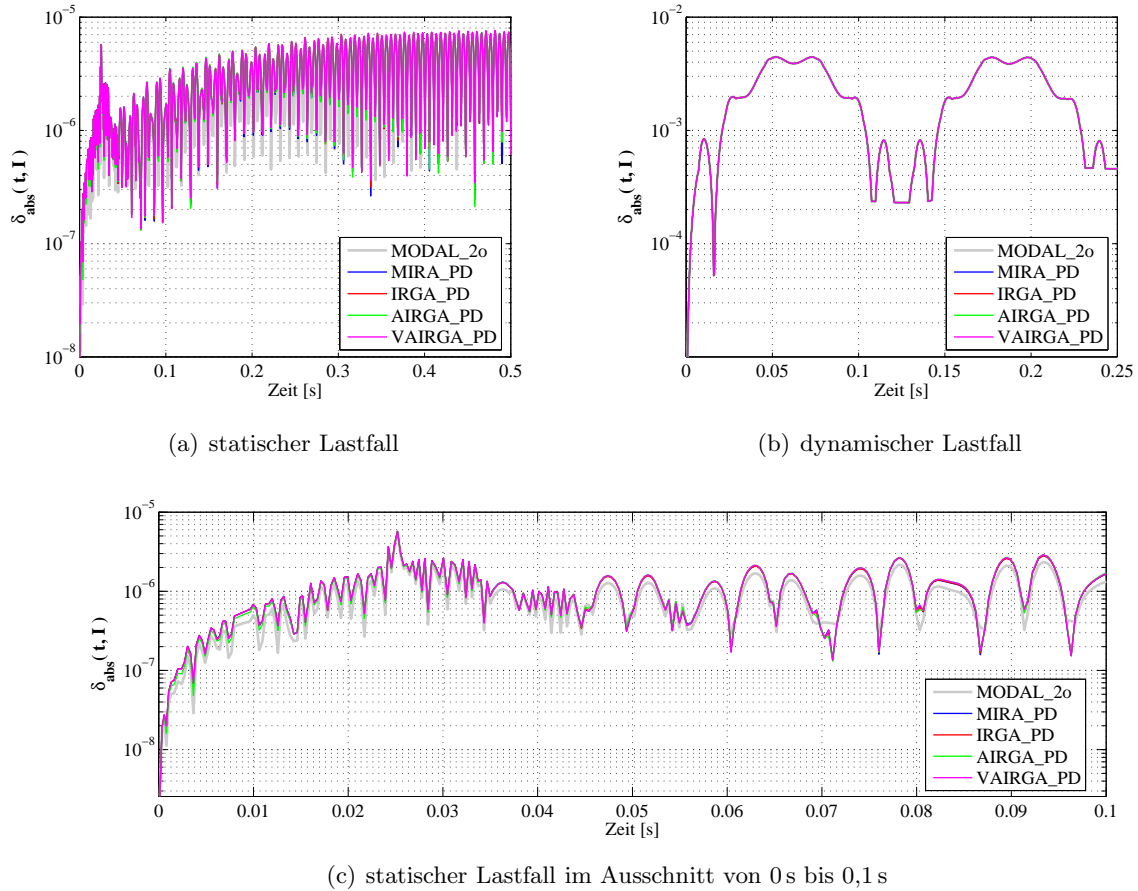


Abbildung 11.3: Absoluter Approximationsfehler $\delta_{abs}(t, \mathcal{I})$ der Simulationsergebnisse für Beispiele eines statischen und eines dynamischen Lastfalles des ersten Beispielmodells, mit den Verfahren angewendet auf das proportional gedämpfte System.

Vorteil auf, dass von Seiten des Anwenders keinerlei Kenntnisse zur optimalen Auswahl geeigneter Moden vorhanden sein müssen.

11.2 Gekoppelte Simulation

Wie bereits im Kapitel 7 dargestellt wurde, ist neben der Simulation von Lastfällen die Regelungssimulation ein wichtiges Hilfsmittel zur Optimierung von Werkzeugmaschinen.

Um die Approximationseigenschaften der, mit den in dieser Arbeit entwickelten Verfahren, reduzierten Systeme in einer Regelsimulation beurteilen zu können, wird hier die Lageregelung als Beispiel einer Regelsimulation betrachtet. Als Modell einer Werkzeugmaschine dient dazu das erste Beispielmodell einer abstrahierten Struktur einer Werkzeugmaschine (siehe Abschnitt 10.1.2.1). Als Vergleichskriterium im Zeitbereich wird dabei der sogenannte Lagesollsprung herangezogen.

Die reduzierten Modelle des mechanischen Systems werden hierfür mit einer kaskadierten Lageregelung, wie sie in Werkzeugmaschinensteuerungen üblich ist und in Abschnitt 7.1 be-

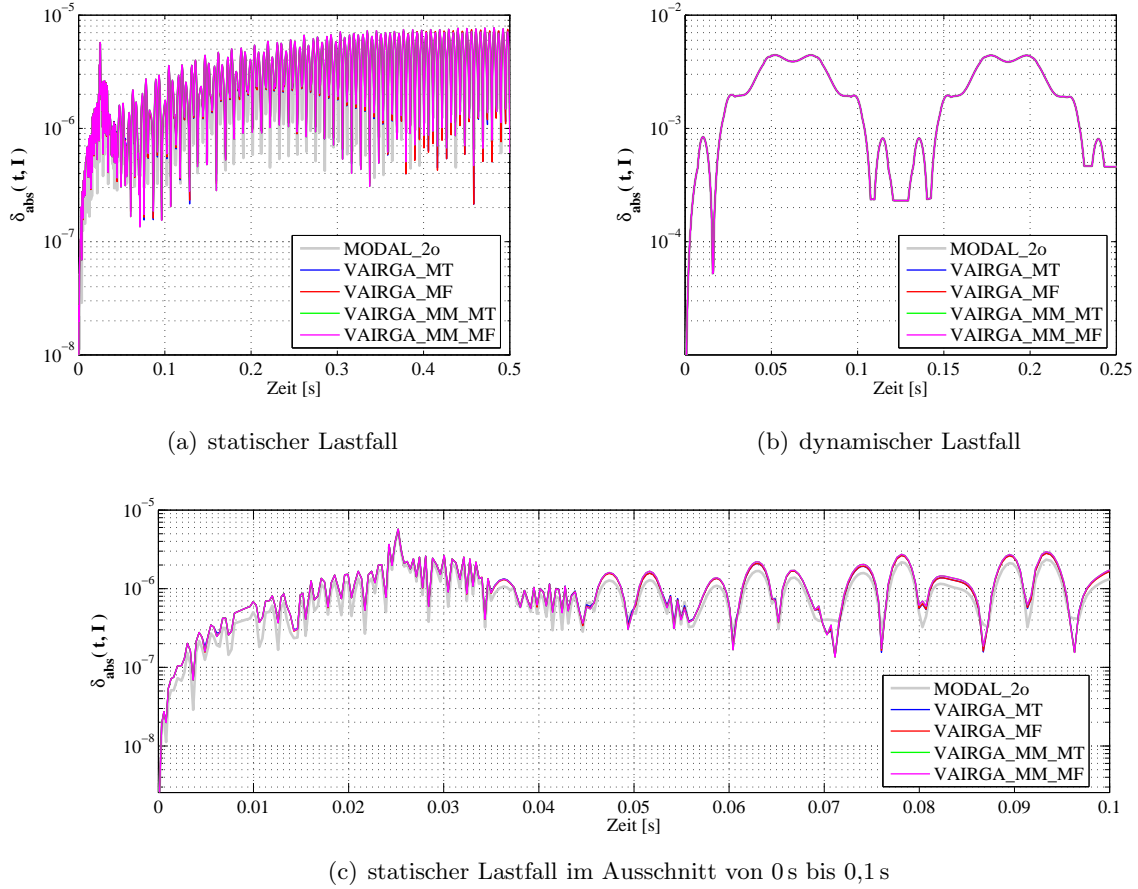


Abbildung 11.4: Absoluter Approximationsfehler $\delta_{abs}(t, \mathcal{I})$ der Simulationsergebnisse für Beispiele eines statischen und eines dynamischen Lastfalles des ersten Beispielsmodells, mit den automatischen Verfahren.

beschrieben wird, in einer Simulationsumgebung gekoppelt. Als Simulationsumgebung wird dabei MATLAB/Simulink[®] verwendet. Als Sollwert wird der Lagereglung ein 1-mm-Sprung vorgegeben, den die Maschine in möglichst kurzer Zeit erreichen soll, wobei der Übertragungspfad vom ersten Eingang (y-Rotation der Motorwelle) zum vierten Ausgang (x-Verlagerung am TCP) betrachtet wird.

Der Entwurf und die Erstellung des kaskadierten Lageregelkreises steht nicht im Fokus dieser Arbeit. Für die Regelsimulation wird deshalb auf eine bereits entwickelte kaskadierte Lageregelung, die am Institut für Werkzeugmaschinen und Betriebswissenschaften (*iwb*) der Technischen Universität München erstellt wurde, zurückgegriffen. Die Abbildung 11.5 stellt die oberste Hierarchieebene dieser kaskadierten Lageregelung für eine einachsige Werkzeugmaschine in der Simulationsumgebung MATLAB/Simulink[®] dar. Bei einer Regelsimulation im Zeitbereich steigt die Rechenzeit aufgrund der Rückkopplung im Regelkreis stark an, so dass eine Lösung für das unreduzierte System bei dieser Größenordnung bereits nicht mehr berechnet werden kann. Als Referenzmodell wird deshalb ein modal reduziertes System der Dimension 400 (MODAL_400) verwendet, das diese Darstellungsform aufweist.

Im Folgenden werden die Ergebnisse des mittels der verschiedenen Krylov-Unterraum-Verfahren

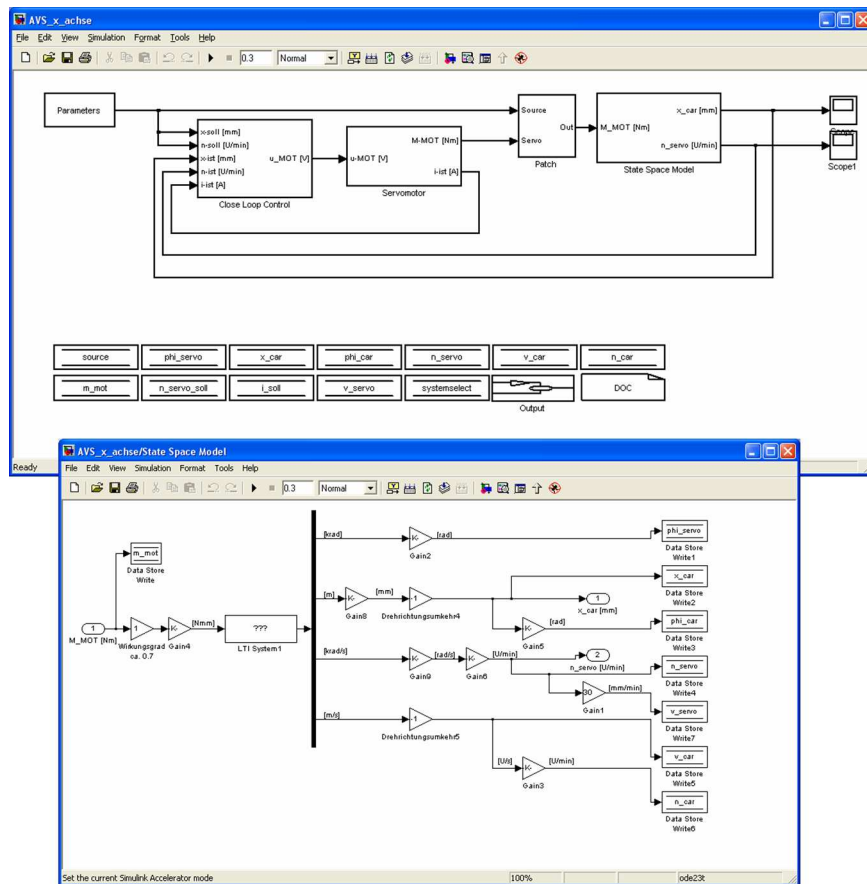


Abbildung 11.5: Oberste Hierarchieebene der kaskadierten Lagregelung² in der Simulationsumgebung MATLAB/Simulink[®]

reduzierten Systems dargestellt. Dazu wird das Originalsystem als System zweiter Ordnung mit den Verfahren MIRA_2o, IRGA_2o, AIRGA_2o und VAIRGA_2o jeweils auf die Dimension $r = 32$ als System zweiter Ordnung reduziert.

Für die Berechnung der reduzierten Systeme werden die Algorithmen der Verfahren (Algorithmus 19, Algorithmus 21 und Algorithmus 22) mit den Anfangsentwicklungsstellen $i \cdot 2\pi$, $166i \cdot 2\pi$, $332i \cdot 2\pi$ und $500i \cdot 2\pi$ sowie $\tau = 10^{-30}$, $tol = 10^{-5}$ und $\varepsilon = (500/3) \cdot 2\pi$ (AIRGA_2o) bzw. $\Delta = 1$ (VAIRGA_2o) verwendet. Die reduzierten Systeme zweiter Ordnung besitzen mit dieser Wahl der Parameter der Reduktionsverfahren die reduzierte Dimension $r = \tilde{m} \cdot \hat{i} \cdot \hat{k} = 4 \cdot 4 \cdot 2 = 32$. Um die Approximationsergebnisse der Verfahren mit den Approximationsergebnissen der automatisierten Verfahren vergleichen zu können, wird wie im vorhergehenden Abschnitt bereits dargestellt wurde, bei der Reduktion mit den automatischen Verfahren durch die geeignete Wahl von ϵ_S und der maximalen reduzierten Dimension das Originalsystem auf Systeme mit der Dimension $r = 32$ reduziert.

Wie bei der Betrachtung von Lastfällen im vorhergehenden Abschnitt, ist bei Krylov-Unterraum-Verfahren zu beachten, dass komplexwertige Projektionsmatrizen V vorliegen, da die Redukti-

²Mit freundlicher Genehmigung des Instituts für Werkzeugmaschinen und Betriebswissenschaften (*iwb*) der TU München

onsverfahren komplexe Entwicklungspunkte verwenden. Um reellwertige Projektionsmatrizen \mathbf{V} zu erhalten, werden die mit den Krylov-Unterraum-Verfahren generierten Matrizen mit der Methode, die in Abschnitt 8.4.4 beschrieben wird, reellwertig gemacht. Die ersten 32 orthogonalen Spaltenvektoren, die mit Hilfe dieser Methode generiert werden, bilden reellwertige Matrizen, die als Projektionsmatrizen \mathbf{V} verwendet werden, um reelle reduzierte Systeme zweiter Ordnung der Dimension $r = 32$ zu erhalten.

Da die Simulationsumgebung MATLAB/Simulink[®] ausschließlich mit Systemen erster Ordnung in der Form einer klassischen Zustandsraumdarstellung (3.12) operiert, werden die reellwertigen reduzierten Systeme zweiter Ordnung mit Hilfe der Transformation (3.5) aus Abschnitt 3.1 in Systeme erster Ordnung transformiert, wodurch sich die Dimension der reduzierten Systeme auf $r = 64$ verdoppelt.

Um die Ergebnisse mit den etablierten Reduktionsmethoden zu vergleichen, werden die mit Hilfe von Krylov-Unterraum-Verfahren reduzierten Systeme einem modal reduzierten System zweiter Ordnung der Dimension 32 (MODAL_2o), welches mit dem FEM-Programm MSC.Nastran[®] generiert und mit Hilfe der Transformation (3.5) in ein System erster Ordnung der Dimension $r = 64$ transformiert wird, sowie dem Referenzmodell der Dimension 400 gegenübergestellt. Die Dimension des modal reduzierten Systems entspricht dabei der Anzahl der Eigenmoden im Intervall $[0 \text{ Hz}, 1464.1 \text{ Hz}]$.

Die Abbildung 11.6 zeigt die absoluten Fehler $\delta_{abs}(t, \{1, 4\})$ der Simulationsergebnisse des beschriebenen Lagesollsprungs der mit den verschiedenen Verfahren reduzierten Systeme.

Resultate

Das Beispiel der Regelsimulation einer Lageregelung zeigt, dass die reduzierten Systeme, die mit den in dieser Arbeit entwickelten Verfahren reduziert werden, gute Approximationsergebnisse liefern. Im Vergleich zum modal reduzierten System zeigt sich ein geringerer absoluter Fehler ($1.5 \cdot 10^{-4}$). Der beobachtete Fehlerverlauf zeigt dabei allerdings ein langsames Einschwingen des Fehlers gegen den Wert 0.

Bei einem Vergleich der Krylov-Unterraum-Verfahren, die auf ungedämpfte bzw. auf proportional gedämpfte Systeme angewendet werden, sind keine Unterschiede im Fehlerverlauf zu erkennen. Beim Vergleich der automatisch reduzierten Systeme ist ein deutlicher Unterschied zwischen den reduzierten Systemen, die die Entwicklungsstellen mit Hilfe des Momentenfehlers (VAIRGA_MM_MT und VAIRGA_MM_MF) und den Verfahren, die die Entwicklungsstellen mit Hilfe des maximalen relativen Fehlers (VAIRGA_MT und VAIRGA_MF) bestimmen, zu erkennen.

Die reduzierten Systeme VAIRGA_MM_MT und VAIRGA_MM_MF weisen annähernd den gleichen Fehlergang wie die mit den Verfahren MIRA_2o, IRGA_2o, AIRGA_2o und VAIRGA_2o reduzierten Systeme auf. Die reduzierten Systeme VAIRGA_MT und VAIRGA_MF besitzen hingegen einen höheren maximalen Approximationsfehler, der bei einem Wert von ca. $2.5 \cdot 10^{-4}$ bzw. $3 \cdot 10^{-4}$ liegt und weisen einen Fehlergang, der sich bei einem Fehler von ca. $2 \cdot 10^{-4}$ bzw. $2.5 \cdot 10^{-4}$ einschwingt, auf.

Die Ursache dafür ist in den ausgewählten Entwicklungsstellen zu sehen, die bei diesen Verfah-

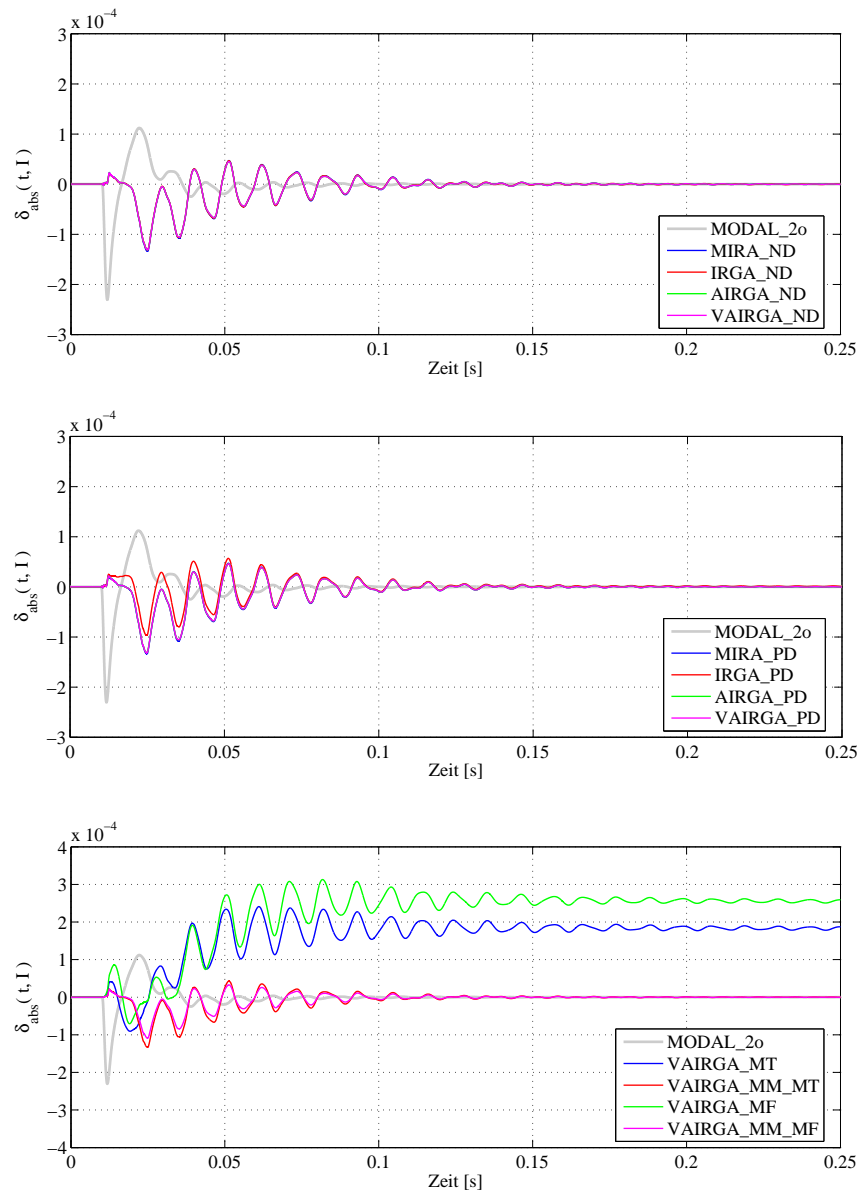


Abbildung 11.6: Absoluter Approximationsfehler $\delta_{abs}(t, 1, 4)$ des Simulationsergebnisses der verschiedenen reduzierten Systeme in Kopplung mit einem kaskadierten Regelkreis.

ren gleichmäßiger über das gewählte Frequenzintervall verteilt ausgewählt werden als dies bei den Verfahren mit Entwicklungsstellenauswahl über die Momentenfehler geschieht. Dieses führt zu einer schlechteren Approximation der Eigenwerte mit betragsmäßig kleinem Realanteil und damit zu einer schlechteren Approximation des Zeitbereichs.

Kapitel 12

Zusammenfassung und Ausblick

Da moderne Werkzeugmaschinen komplexe mechatronische Produktionssysteme sind, deren Entwicklung unter großem Innovations-, Zeit- und Kostendruck stattfindet, erlangen Techniken, die den Entwicklungsprozess unterstützen, zunehmend an Bedeutung.

Zur Absicherung der Realisierbarkeit neuer Funktionen und zur Überprüfung der Erreichbarkeit von Qualitäts- und Leistungsanforderungen einer Werkzeugmaschine kommen heutzutage neue Techniken zum Einsatz. Die Simulationstechnik stellt in diesem Zusammenhang eine Schlüsseltechnologie zur frühzeitigen Verifikation des Verhaltens von Produktionsanlagen dar.

Ein wichtiger Ansatz zur Abschätzung der erreichbaren Leistungsfähigkeit und Genauigkeit von NC-Maschinenentwürfen ist die Kopplung von FEM-Modellen der mechanischen Struktur mit der regelungstechnischen Simulation der elektrischen Antriebssysteme. Dabei werden Entwurfsmodelle der mechanischen und elektrischen Konstruktion sowie der Steuerungstechnik mit geringfügigem Zusatzaufwand für die Modelldatenverarbeitung in ein mechatronisches Gesamtmodell integriert.

Für die Simulation des Verhaltens einer Maschine mit einer guten Auflösung müssen sehr hochdimensionale Modelle verwendet werden. Für eine effiziente Berechnung des Gesamtsystems, ist es dann notwendig, dass die Anzahl der Freiheitsgrade, die durch die Diskretisierung der Finite-Elemente-Methode bestimmt wird, deutlich reduziert wird. Hierfür werden Verfahren der sogenannten modalen Reduktion verwendet, die auf der Lösung des Eigenwertproblems beruhen, um damit das Originalsystem mit Hilfe der Modalmatrix auf einen deutlich kleineren, modalen Raum zu projizieren.

Da mit diesen Verfahren eine optimale Approximation lediglich durch die Berechnung sämtlicher Eigenwerte erreicht werden kann, was zu sehr langen Berechnungszeiten führt, werden lediglich vom Anwender gewählte oder die niederfrequenten Eigenwerte für die Ordnungsreduktion verwendet. Dies hat zur Folge, dass die entstehenden reduzierten Modelle zum Teil erhebliche Fehler im statischen Bereich aufweisen. Zudem berücksichtigen die modalen Verfahren lediglich das System alleinstehend, es erfolgt demnach keine Beachtung der Systemein- bzw. -ausgänge und es existieren keine Schranken für den Approximationsfehler.

Neue mathematische Ansätze zur Modellreduktion, welche gegenüber den konventionellen Methoden zahlreiche Vorteile aufweisen, lassen ein erhebliches Potenzial erkennen das Originalsy-

stem besser approximieren zu können. Zudem lassen sich diese automatisieren und weisen über ein breiteres Frequenzband sehr gute Approximationseigenschaften auf. Diese Verfahren weisen auch die nötige Effizienz auf, um zukünftige Maschinengenerationen bereits am virtuellen Prototypen zuverlässig testen zu können.

Neben dem erkennbaren Potential der mathematischen Verfahren weisen diese jedoch noch Probleme bei der optimalen Auslegung der Algorithmen auf. Dazu gehört die Wahl der verwendeten Entwicklungsstellen und der Stufen der verwendeten Krylov-Unterräume, die Implementierung einer praktisch anwendbaren Methode zur Fehlerschätzung des reduzierten Systems sowie einer Methode zur Automatisierung des Reduktionsprozesses.

In dieser Arbeit wurden angepasste mathematische Verfahren zur automatisierten Modellreduktion der Finite-Elemente-Modelle, die durch die Modellierung rotatorischer Antriebssysteme mit Kugelgewindetrieb im Verbund mit dem Maschinengestell resultieren, entwickelt und auf die Prozesstauglichkeit hinsichtlich der Kopplung mit einem Bearbeitungsprozessmodell validiert.

Da das reduzierte System zur weiteren Verwendung in der Simulation von Werkzeugmaschinen spezielle Anforderungen zu erfüllen hat, wurden in der vorliegenden Arbeit verschiedene Reduktionsverfahren entwickelt, die reduzierte Systeme generieren, die diesen Anforderungen genügen. Dazu wurden zum einen neue Methoden entwickelt und zum anderen bestehende Methoden angepasst.

Dazu wird ein Verfahren vorgestellt, das vom Autor in seinen Grundzügen bereits in [57, 58] veröffentlicht wurde und *modifizierter iterativer rationaler Krylov-Algorithmus* (MIRKA) genannt wird. Dieses Verfahren basiert auf einer iterativen Bestimmung der Entwicklungsstellen, die an die speziellen hier betrachteten Modelle rotatorischer Antriebssysteme mit Kugelgewindetrieb angepasst ist. Dieses Verfahren kann sowohl auf Systeme erster Ordnung als auch auf Systeme zweiter Ordnung angewendet werden und basiert auf dem Lanczos-Algorithmus (MIRL) oder auf dem Arnoldi-Algorithmus (MIRA).

Das Verfahren MIRKA dient als Grundlage zur Entwicklung eines effizienten Verfahrens zur Berechnung von MIMO-Systemen. Dazu wird in dieser Arbeit ein neues Verfahren vorgestellt, das auf globalen Krylov-Unterräumen basiert und die globale Arnoldi-Methode verwendet, die sich durch hohe Effizienz bei der Berechnung der Reduktion von Systemen mit mehreren Ein- und Ausgängen auszeichnet. Dieses Verfahren wird hier zur Reduktion von Systemen zweiter Ordnung verwendet und *iterativer rationaler globaler Arnoldi-Algorithmus für Systeme zweiter Ordnung* (IRGA_2o) genannt.

Um eine optimale Wahl der Krylov-Unterräume zu den iterativ bestimmten Entwicklungsstellen zu gewährleisten, wurde das Verfahren IRGA_2o um eine adaptive Bestimmung der Stufen der Krylov-Unterräume zu den korrespondierenden Entwicklungsstellen erweitert. Die adaptive Bestimmung der Stufen der Krylov-Unterräume zu den jeweiligen Entwicklungsstellen basiert dabei auf der Methode, die in [39] vorgestellt wurde und die hier angepasst auf Systeme zweiter Ordnung zur Anwendung kommt. Das aus der Kombination des Verfahrens IRGA_2o und der adaptiven Bestimmung der Stufen der Krylov-Unterräume resultierende Verfahren wird hier *adaptiver iterativer rationaler globaler Arnoldi-Algorithmus für Systeme zweiter Ordnung* (AIRGA_2o) bzw. *vollständiger adaptiver iterativer rationaler globaler Arnoldi-Algorithmus für*

Systeme zweiter Ordnung (VAIRGA_2o) genannt.

Im weiteren wurde das Verfahren VAIRGA_2o um eine Methode zur Fehlerschätzung erweitert. Die hier verwendete Methode basiert dabei auf der Auswertung des Approximationsfehlers an den iterativ bestimmten Entwicklungsstellen. Mit dieser Methode ist die Fehlerschätzung effizient berechenbar und an keine Fehlernorm gebunden.

Mit Hilfe der Fehlerschätzung ist es möglich, eine Automatisierung des Reduktionsprozesses mit Hilfe des Verfahrens VAIRGA_2o zu verwirklichen. Als Eingabeparameter der automatisierten Verfahren, werden nur die Vorgabe des zu approximierenden Frequenzbereichs, die Anzahl der Anfangsentwicklungsstellen und die maximale Dimension des reduzierten Modells benötigt. Der Anwender erhält als Ausgabeinformation den geschätzten maximalen Approximationsfehler des reduzierten Systems. Zudem erlauben diese Verfahren eine Modellreduktion ohne Benutzerinteraktion während des Reduktionsprozesses.

Die in dieser Arbeit entwickelten Verfahren wurden anhand von zwei Modellen validiert. Dabei zeigte die Validierung der verschiedenen Verfahren die Anwendbarkeit der entwickelten Verfahren zur Reduktion der in der Arbeit betrachteten Modelle von Werkzeugmaschinen. Bei der Validierung der automatisierten Verfahren konnten mit den Verfahren, die die Auswahl der Entwicklungsstellen über die Auswertung des maximalen Momentenfehlers treffen, gute Reduktionsergebnisse erzielt werden.

In [28] werden die Verfahren auf ein Beispielmittel der Dimension 187.026 angewendet. Dabei zeigt sich, dass die dargestellten Reduktionsergebnisse der in dieser Arbeit entwickelten Verfahren auch auf Modelle mit höheren Freiheitsgraden übertragen werden können. Diese Ergebnisse wurden zudem auf der VDI-Mechatroniktagung 2011 in Dresden präsentiert.

Zusammenfassend konnten in dieser Arbeit somit durch die Entwicklung neuer Verfahren, die auf Krylov-Unterräumen basieren, die folgenden Ziele erreicht werden:

1. Mit Hilfe des globalen Arnoldi Algorithmus konnte ein effizientes Krylov-Unterraum-Verfahren auf MIMO-Systeme zweiter Ordnung angewendet werden.
2. Die Anwendbarkeit mathematischer Verfahren zur Modellreduktion von Strukturmodellen konnte gezeigt und mit modal reduzierten Modellen verglichen werden.
3. Die Prozessstauglichkeit hinsichtlich der Kopplung mit einem Bearbeitungsprozessmodell konnte gezeigt werden.
4. Mit den Verfahren AIRGA_2o und VAIRGA_2o wurde ein Krylov-Unterraum-Verfahren vorgestellt, das eine Reduktion von Systemen zweiter Ordnung durch eine iterative Bestimmung der Entwicklungsstellen und einer adaptiven Wahl der Stufe der verwendeten Krylov-Unterräume an diesen Stellen durchführt. Der Anwender dieses Verfahrens wird somit der Entscheidungen bezüglich der optimalen Entwicklungsstellenwahl und der Bestimmung der optimalen Stufen der Krylov-Unterräume an diesen Stellen enthoben.
5. Mit Hilfe eines an das Verfahren VAIRGA_2o angepassten Fehlerschätzers konnten automatische Modellreduktionsverfahren entwickelt werden. Als Eingabeparameter der automatisierten Verfahren, werden nur die Vorgabe des zu approximierenden Frequenzbereichs,

die Anzahl der Anfangsentwicklungsstellen und die maximale Dimension des reduzierten Modells benötigt. Der Anwender erhält als Ausgabeinformation den geschätzten maximalen Approximationsfehler des reduzierten Systems.

Die in der Praxis verwendeten Modelle von Werkzeugmaschinen weisen gewöhnlich höhere Freiheitsgrade auf, als die zur Validierung der entwickelten Verfahren verwendeten Beispielm Modelle. Deshalb dienen die in dieser Arbeit vorgeschlagenen Verfahren als theoretische Grundlage für eine automatische Reduktion von Strukturmodellen. Für eine Anwendung in der Praxis sind die Algorithmen der vorgestellten Verfahren jedoch noch nicht ausgereift, um ihr volles Potential ausschöpfen zu können.

Zur Ausschöpfung des vollen Potentials der Verfahren kann und sollte über weitere Optimierungen speziell bei der Implementierung der Verfahren nachgedacht werden, auf die im Rahmen dieser Arbeit nicht weiter eingegangen werden konnte.

Folgende Ansatzpunkte bieten sich zur weiteren Optimierung der vorgeschlagenen Verfahren an:

1. Die Verfahren wurden in MATLAB® implementiert. Weitere Effizienzsteigerungen sollten sich durch eine Implementierung in C, C++ oder Fortran ergeben.
2. Die höchsten numerischen Kosten der Reduktionsverfahren auf Basis von Krylov-Unterräumen verursacht das Lösen der auftretenden Gleichungssysteme. An dieser Stelle muss die Verwendung von iterativen Gleichungslösern noch weiter untersucht werden.
3. Mit den vorgeschlagenen Verfahren sollten sich durch Parallelisierung der Algorithmen deutliche Effizienzsteigerungen ergeben.

Anhang

Im vorliegenden Anhang werden die Approximationsfehler der reduzierten Beispielm Modelle unterschiedlicher Dimensionen dargestellt. Im ersten Teil des Anhanges (A.1) erfolgt die Darstellung der Approximationsfehler des ersten Beispielm Modells, das auf die Dimensionen 16, 24, 32 und 40 reduziert wurde. Im zweiten Teil des Anhanges (A.2) erfolgt die Darstellung der Approximationsfehler des zweiten Beispielm Modells, das auf die Dimensionen 32, 40, 48 und 56 reduziert wurde.

A.1 Erstes Beispielm Modell reduziert auf die Dimensionen 16, 24, 32 und 40

Im folgenden sind die Approximationsergebnisse im Frequenzbereich des mit den verschiedenen Verfahren reduzierten ersten Beispielm Modells der Dimension 16, 24, 32 und 40 dargestellt.

Die Algorithmen des Verfahrens MIRA_ND (Algorithmus 19) und des Verfahrens IRGA_ND (Algorithmus 21) verwenden für die Stufe des Krylov-Unterraumes $k_i = [\mathbf{k}]_i = 2$ an jeder Entwicklungsstelle $s_i = [\mathbf{s}]_i$, $i = 1, \dots, \hat{i}$. Die Dimension der reduzierten Systeme r ergibt sich durch die Anzahl der Anfangsentwicklungsstellen \hat{i} , der Stufe des Krylov-Unterraumes k_i an jeder Entwicklungsstelle s_i und der Anzahl der (linear unabhängigen) Eingänge \tilde{m} damit zu $r = \tilde{m} \cdot \hat{i} \cdot \hat{k}$. Zudem wird für alle Verfahren zur Berechnung der reduzierten Systeme mit unterschiedlichen Dimensionen $tol = 10^{-5}$ und $f_{max} = 500$ verwendet.

Approximationsfehler reduzierter Systeme der Dimension 16

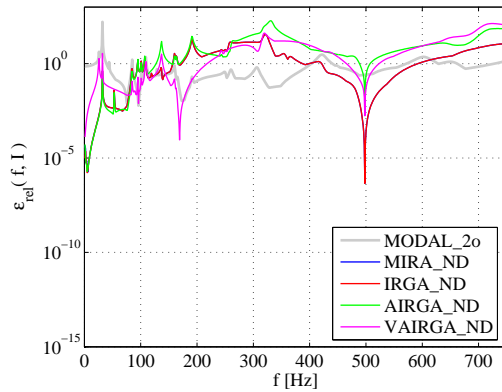
Die Algorithmen der Reduktionsverfahren (Algorithmus 19, Algorithmus 21 und Algorithmus 22) werden mit den Anfangsentwicklungsstellen $\iota \cdot 2\pi$ und $500\iota \cdot 2\pi$ sowie $\Delta = 500 \cdot 2\pi$ (AIRGA_2o) bzw. $\Delta = 0$ (VAIRGA_2o) verwendet. Durch die Anzahl der Anfangsentwicklungsstellen $\hat{i} = 2$, der Stufe des Krylov-Unterraumes $k_i = 2$ an jeder Entwicklungsstelle s_i und der Anzahl der (linear unabhängigen) Eingänge $\tilde{m} = 4$, ergibt sich die Dimension der reduzierten Systeme damit zu $r = \tilde{m} \cdot \hat{i} \cdot \hat{k} = 4 \cdot 2 \cdot 2 = 16$.

Bei der Betrachtung der Approximationsergebnisse der reduzierten Systeme werden die Krylov-Unterraum reduzierten Modelle mit reduzierter Dimension 16 einem modal reduzierten Modell der gleichen Dimension (MODAL_2o) gegenüber gestellt. Die Dimension des modal reduzierten Systems entspricht der Anzahl der Eigenmoden im Intervall $[0 \text{ Hz}, 685.91 \text{ Hz}]$.

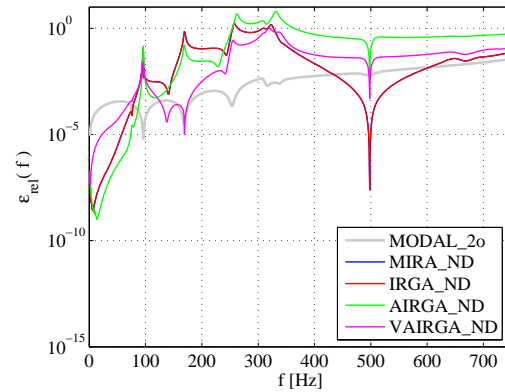
Die Ergebnisse der verschiedenen Reduktionsverfahren sind in der Tabelle A.1 zusammengefasst. Die Abbildungen A.1 und A.2 geben den Verlauf der relativen Fehler (10.3) und (10.5) der verschiedenen reduzierten Modelle wieder.

Tabelle A.1: Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 16

Verfahren	Moden	Zeit [s]	Anzahl Iterationen	Fehler $\epsilon_{\max}(\mathcal{I})$	Fehler ϵ_{\max}
MODAL_2o	16	–	–	163.15	$7.36 \cdot 10^{-3}$
Verfahren mit Anwendung auf das ungedämpfte System					
MIRA_ND	–	3.56	2	34.61	1.66
IRGA_ND	–	2.64	2	34.61	1.66
AIRGA_ND	–	3.30	2	188.06	6.12
VAIRGA_ND	–	11.99	2	41.38	1.07
Verfahren mit Anwendung auf das proportional gedämpfte System					
MIRA_PD	–	4.00	2	34.36	1.62
IRGA_PD	–	3.84	2	34.36	1.62
AIRGA_PD	–	4.78	2	183.55	5.76
VAIRGA_PD	–	14.09	2	40.88	1.06



(a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$



(b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung A.1: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND der Dimension 16, angewendet auf das ungedämpfte erste Beispielmodell

Approximationsfehler reduzierter Systeme der Dimension 24

Die Algorithmen der Reduktionsverfahren (Algorithmus 19, Algorithmus 21 und Algorithmus 22) werden mit den Anfangsentwicklungsstellen $i \cdot 2\pi$, $250i \cdot 2\pi$ und $500i \cdot 2\pi$ sowie $\Delta = 500 \cdot 2\pi$

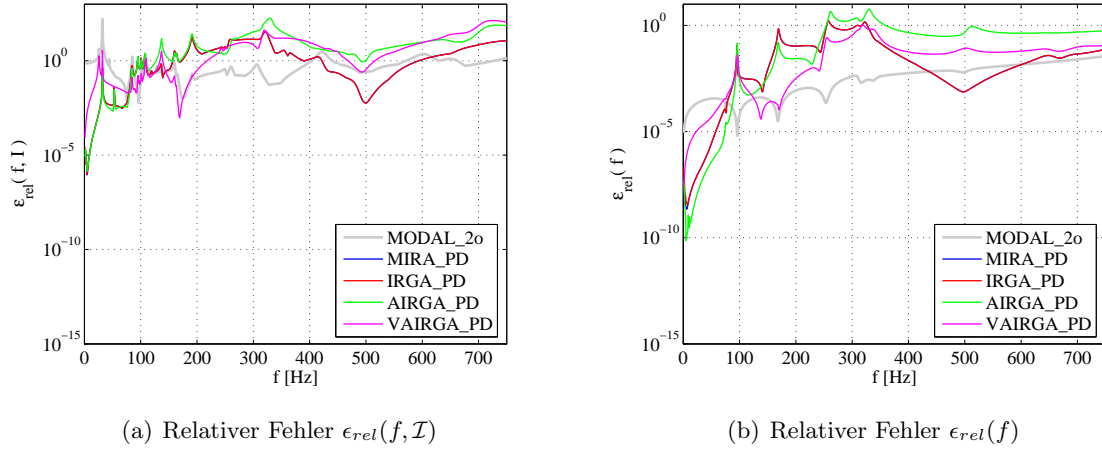


Abbildung A.2: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD der Dimension 16, angewendet auf das proportional gedämpfte erste Beispielmodell

(AIRGA_2o) bzw. $\Delta = 0$ (VAIRGA_2o) verwendet. Durch die Anzahl der Anfangsentwicklungsstellen $\hat{i} = 3$, der Stufe des Krylov-Unterraumes $k_i = 2$ an jeder Entwicklungsstelle s_i und der Anzahl der (linear unabhängigen) Eingänge $\tilde{m} = 4$, ergibt sich die Dimension der reduzierten Systeme damit zu $r = \tilde{m} \cdot \hat{i} \cdot \hat{k} = 4 \cdot 3 \cdot 2 = 24$.

Bei der Betrachtung der Approximationsergebnisse der reduzierten Systeme werden die Krylov-Unterraum reduzierten Modelle mit reduzierter Dimension 24 einem modal reduzierten Modell der gleichen Dimension (MODAL_2o) gegenüber gestellt. Die Dimension des modal reduzierten Systems entspricht der Anzahl der Eigenmoden im Intervall $[0 \text{ Hz}, 1023.4 \text{ Hz}]$.

Die Ergebnisse der verschiedenen Reduktionsverfahren sind in der Tabelle A.2 zusammengefasst. Die Abbildungen A.3 und A.4 geben den Verlauf der relativen Fehler (10.3) und (10.5) der verschiedenen reduzierten Modelle wieder.

Approximationsfehler reduzierter Systeme der Dimension 32

Die Algorithmen der Reduktionsverfahren (Algorithmus 19, Algorithmus 21 und Algorithmus 22) werden mit den Anfangsentwicklungsstellen $i \cdot 2\pi$, $166i \cdot 2\pi$, $332i \cdot 2\pi$ und $500i \cdot 2\pi$ (AIRGA_2o) bzw. $\Delta = 0$ (VAIRGA_2o) verwendet. Durch die Anzahl der Anfangsentwicklungsstellen $\hat{i} = 4$, der Stufe des Krylov-Unterraumes $k_i = 2$ an jeder Entwicklungsstelle s_i und der Anzahl der (linear unabhängigen) Eingänge $\tilde{m} = 4$, ergibt sich die Dimension der reduzierten Systeme damit zu $r = \tilde{m} \cdot \hat{i} \cdot \hat{k} = 4 \cdot 4 \cdot 2 = 32$.

Bei der Betrachtung der Approximationsergebnisse der reduzierten Systeme werden die Krylov-Unterraum reduzierten Modelle mit reduzierter Dimension 32 einem modal reduzierten Modell der gleichen Dimension (MODAL_2o) gegenüber gestellt. Die Dimension des modal reduzierten Systems entspricht der Anzahl der Eigenmoden im Intervall $[0 \text{ Hz}, 1464.1 \text{ Hz}]$.

Die Ergebnisse der verschiedenen Reduktionsverfahren sind in der Tabelle A.3 zusammengefasst. Die Abbildungen A.5 und A.6 geben den Verlauf der relativen Fehler (10.3) und (10.5) der

Tabelle A.2: Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 24

Verfahren	Moden	Zeit [s]	Anzahl Iterationen	Fehler $\epsilon_{max}(\mathcal{I})$	Fehler ϵ_{max}
MODAL_2o	24	–	–	71.81	$4.20 \cdot 10^{-3}$
Verfahren mit Anwendung auf das ungedämpfte System					
MIRA_ND	–	4.36	2	2.22	$0.11 \cdot 10^{-3}$
IRGA_ND	–	4.03	2	1.52	$1.22 \cdot 10^{-3}$
AIRGA_ND	–	5.76	2	1.28	$1.44 \cdot 10^{-3}$
VAIRGA_ND	–	17.80	2	2.09	$0.20 \cdot 10^{-3}$
Verfahren mit Anwendung auf das proportional gedämpfte System					
MIRA_PD	–	6.45	2	2.21	$0.11 \cdot 10^{-3}$
IRGA_PD	–	5.95	2	2.12	$0.24 \cdot 10^{-3}$
AIRGA_PD	–	8.33	2	1.18	$2.92 \cdot 10^{-3}$
VAIRGA_PD	–	20.42	2	2.23	$0.44 \cdot 10^{-3}$

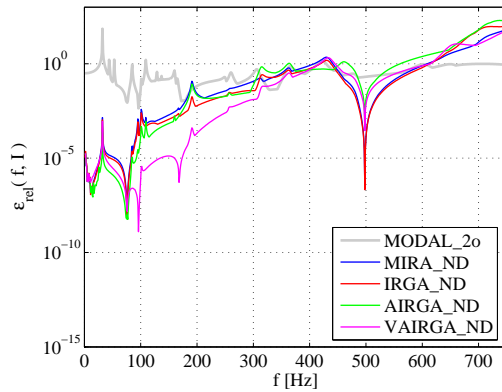
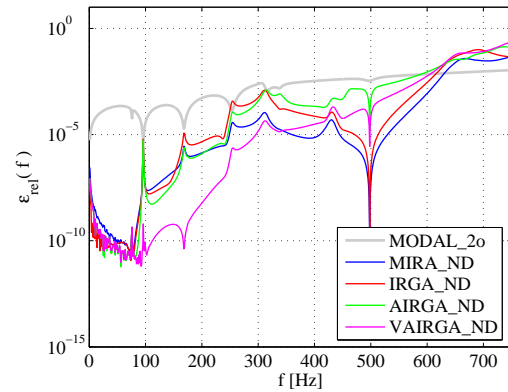
(a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$ (b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung A.3: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND der Dimension 24, angewendet auf das ungedämpfte erste Beispielmodell

verschiedenen reduzierten Modelle wieder.

Approximationsfehler reduzierter Systeme der Dimension 40

Die Algorithmen der Reduktionsverfahren (Algorithmus 19, Algorithmus 21 und Algorithmus 22) werden mit den Anfangsentwicklungsstellen $\iota \cdot 2\pi$, $125.75\iota \cdot 2\pi$, $250.5\iota \cdot 2\pi$, $375.25\iota \cdot 2\pi$ und $500\iota \cdot 2\pi$ sowie $\Delta = 500 \cdot 2\pi$ (AIRGA_2o) bzw. $\Delta = 0$ (VAIRGA_2o) verwendet. Durch die Anzahl der Anfangsentwicklungsstellen $\hat{\iota} = 5$, der Stufe des Krylov-Unterraumes $k_i = 2$ an jeder

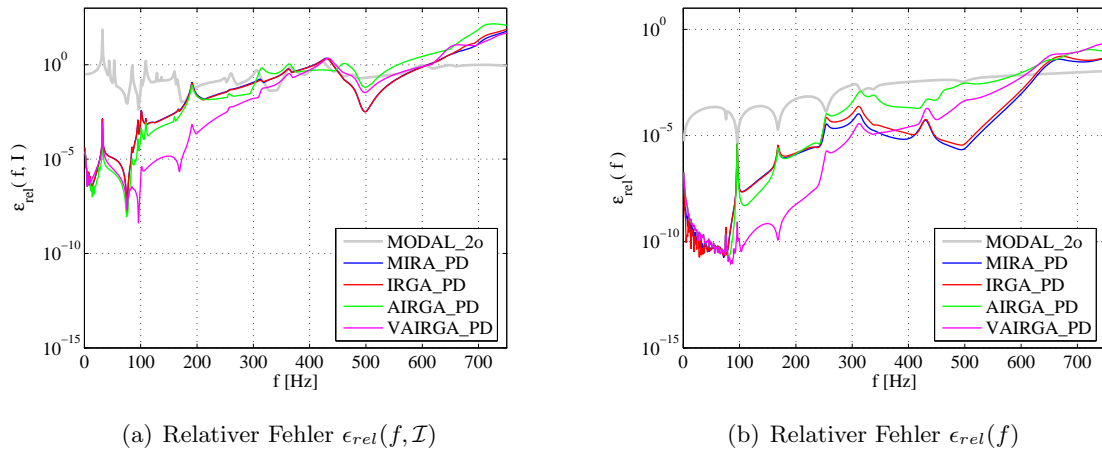


Abbildung A.4: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD der Dimension 24, angewendet auf das proportional gedämpfte erste Beispielmmodell

Tabelle A.3: Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 32

Verfahren	Moden	Zeit [s]	Anzahl Iterationen	Fehler $\epsilon_{max}(\mathcal{I})$	Fehler ϵ_{max}
MODAL_2o	32	—	2	20.56	$3.84 \cdot 10^{-3}$
Verfahren mit Anwendung auf das ungedämpfte System					
MIRA_ND	—	6.25	2	$11.82 \cdot 10^{-3}$	$0.86 \cdot 10^{-6}$
IRGA_ND	—	5.64	2	$0.24 \cdot 10^{-3}$	$17.06 \cdot 10^{-6}$
AIRGA_ND	—	9.97	2	$10.89 \cdot 10^{-3}$	$9.73 \cdot 10^{-6}$
VAIRGA_ND	—	23.13	2	$3.63 \cdot 10^{-3}$	$2.44 \cdot 10^{-6}$
Verfahren mit Anwendung auf das proportional gedämpfte System					
MIRA_PD	—	9.05	2	$6.14 \cdot 10^{-3}$	$2.13 \cdot 10^{-6}$
IRGA_PD	—	8.25	2	$0.47 \cdot 10^{-3}$	$10.69 \cdot 10^{-6}$
AIRGA_PD	—	14.61	2	$24.73 \cdot 10^{-3}$	$21.81 \cdot 10^{-6}$
VAIRGA_PD	—	27.95	2	$0.67 \cdot 10^{-3}$	$2.86 \cdot 10^{-6}$

Entwicklungsstelle s_i und der Anzahl der (linear unabhängigen) Eingänge $\tilde{m} = 4$, ergibt sich die Dimension der reduzierten Systeme damit zu $r = \tilde{m} \cdot \hat{i} \cdot \hat{k} = 4 \cdot 5 \cdot 2 = 40$.

Bei der Betrachtung der Approximationsergebnisse der reduzierten Systeme werden die Krylov-Unterraum reduzierten Modelle mit reduzierter Dimension 40 einem modal reduzierten Modell der gleichen Dimension (MODAL_2o) gegenüber gestellt. Die Dimension des modal reduzierten Systems entspricht der Anzahl der Eigenmoden im Intervall $[0 \text{ Hz}, 1839.7 \text{ Hz}]$.

Die Ergebnisse der verschiedenen Reduktionsverfahren sind in der Tabelle A.4 zusammengefasst. Die Abbildungen A.7 und A.8 geben den Verlauf der relativen Fehler (10.3) und (10.5) der

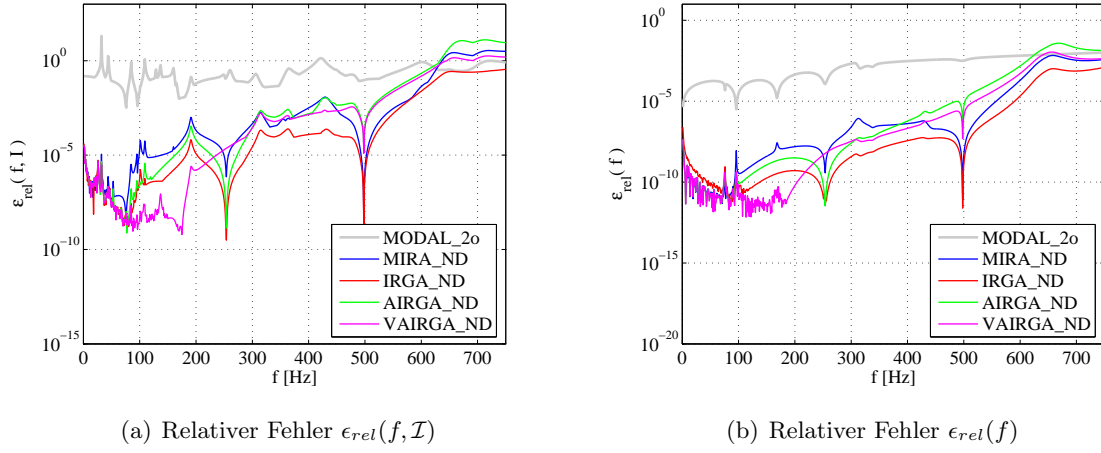


Abbildung A.5: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND der Dimension 32, angewendet auf das ungedämpfte erste Beispielmodell

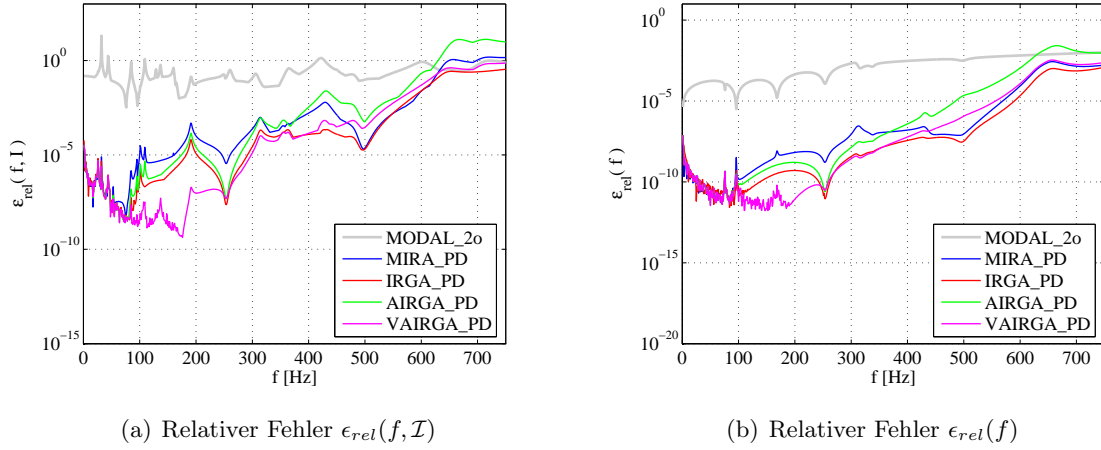


Abbildung A.6: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD der Dimension 32, angewendet auf das proportional gedämpfte erste Beispielmodell

verschiedenen reduzierten Modelle wieder.

Tabelle A.4: Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 40

Verfahren	Moden	Zeit [s]	Anzahl Iterationen	Fehler $\epsilon_{max}(\mathcal{I})$	Fehler ϵ_{max}
MODAL_2o	40	–	–	2129.80	$4.57 \cdot 10^{-3}$
Verfahren mit Anwendung auf das ungedämpfte System					
MIRA_ND	–	8.45	2	$1.51 \cdot 10^{-3}$	$4.83 \cdot 10^{-6}$
IRGA_ND	–	7.38	2	$0.76 \cdot 10^{-3}$	$1.25 \cdot 10^{-6}$
AIRGA_ND	–	13.08	2	$0.75 \cdot 10^{-3}$	$0.21 \cdot 10^{-6}$
VAIRGA_ND	–	29.89	2	$0.75 \cdot 10^{-3}$	$0.21 \cdot 10^{-6}$
Verfahren mit Anwendung auf das proportional gedämpfte System					
MIRA_PD	–	12.34	2	$0.26 \cdot 10^{-3}$	$4.57 \cdot 10^{-6}$
IRGA_PD	–	10.77	2	$0.81 \cdot 10^{-3}$	$16.07 \cdot 10^{-6}$
AIRGA_PD	–	19.63	2	$0.43 \cdot 10^{-3}$	$5.98 \cdot 10^{-6}$
VAIRGA_PD	–	36.09	2	$0.43 \cdot 10^{-3}$	$5.98 \cdot 10^{-6}$

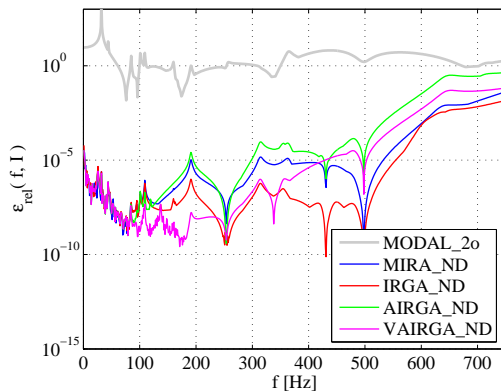
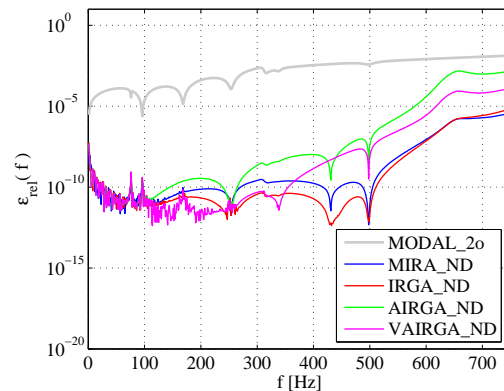
(a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$ (b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung A.7: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND der Dimension 40, angewendet auf das ungedämpfte erste Beispielmodell

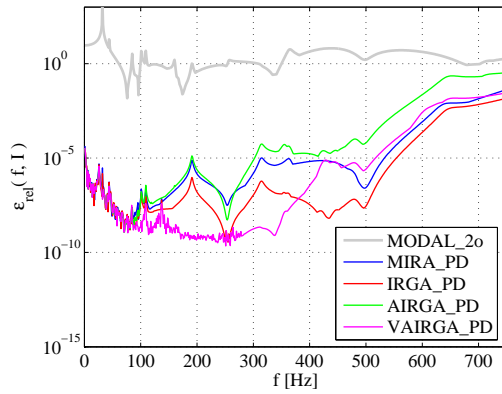
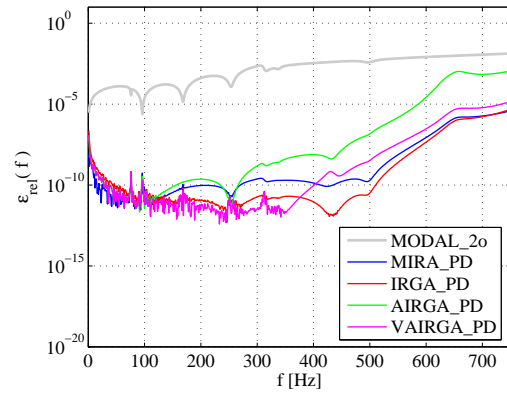
(a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$ (b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung A.8: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD der Dimension 40, angewendet auf das proportional gedämpfte erste Beispielmodell

A.2 Zweites Beispielmodell reduziert auf die Dimensionen 32, 40, 48 und 56

Im folgenden sind die Approximationsergebnisse im Frequenzbereich des mit den verschiedenen Verfahren reduzierten zweiten Beispielmodells der Dimension 32, 40, 48 und 56 dargestellt.

Die Algorithmen des Verfahrens MIRA_ND (Algorithmus 19) und des Verfahrens IRGA_ND (Algorithmus 21) verwenden für die Stufe des Krylov-Unterraumes $k_i = [\mathbf{k}]_i = 2$ an jeder Entwicklungsstelle $s_i = [\mathbf{s}]_i$, $i = 1, \dots, \hat{i}$. Die Dimension der reduzierten Systeme r ergibt sich durch die Anzahl der Anfangsentwicklungsstellen \hat{i} , der Stufe des Krylov-Unterraumes k_i an jeder Entwicklungsstelle s_i und der Anzahl der (linear unabhängigen) Eingänge \tilde{m} damit zu $r = \tilde{m} \cdot \hat{i} \cdot \hat{k}$. Zudem wird für alle Verfahren zur Berechnung der reduzierten Systeme mit unterschiedlichen Dimensionen $tol = 10^{-5}$ und $f_{max} = 500$ verwendet.

Approximationsfehler reduzierter Systeme der Dimension 32

Die Algorithmen der Reduktionsverfahren (Algorithmus 19, Algorithmus 21 und Algorithmus 22) werden mit den Anfangsentwicklungsstellen $\iota \cdot 2\pi$, $167\iota \cdot 2\pi$, $332\iota \cdot 2\pi$ und $500\iota \cdot 2\pi$ sowie $\Delta = 500 \cdot 2\pi$ (AIRGA_2o) bzw. $\Delta = 0$ (VAIRGA_2o) verwendet. Durch die Anzahl der Anfangsentwicklungsstellen $\hat{i} = 4$, der Stufe des Krylov-Unterraumes $k_i = 2$ an jeder Entwicklungsstelle s_i und der Anzahl der (linear unabhängigen) Eingänge $\tilde{m} = 4$, ergibt sich die Dimension der reduzierten Systeme damit zu $r = \tilde{m} \cdot \hat{i} \cdot \hat{k} = 4 \cdot 4 \cdot 2 = 32$.

Bei der Betrachtung der Approximationsergebnisse der reduzierten Systeme werden die Krylov-Unterraum reduzierten Modelle mit reduzierter Dimension 32 einem modal reduzierten Modell der gleichen Dimension (MODAL_2o) gegenüber gestellt. Die Dimension des modal reduzierten Systems entspricht der Anzahl der Eigenmoden im Intervall $[0 \text{ Hz}, 962.86 \text{ Hz}]$.

Die Ergebnisse der verschiedenen Reduktionsverfahren sind in der Tabelle A.5 zusammengefasst. Die Abbildungen A.9 und A.10 geben den Verlauf der relativen Fehler (10.3) und (10.5) der verschiedenen reduzierten Modelle wieder.

Approximationsfehler reduzierter Systeme der Dimension 40

Die Algorithmen der Reduktionsverfahren (Algorithmus 19, Algorithmus 21 und Algorithmus 22) werden mit den Anfangsentwicklungsstellen $\iota \cdot 2\pi$, $125.75\iota \cdot 2\pi$, $250.5\iota \cdot 2\pi$, $375.25\iota \cdot 2\pi$ und $500\iota \cdot 2\pi$ sowie $\Delta = 500 \cdot 2\pi$ (AIRGA_2o) bzw. $\Delta = 0$ (VAIRGA_2o) verwendet. Durch die Anzahl der Anfangsentwicklungsstellen $\hat{i} = 5$, der Stufe des Krylov-Unterraumes $k_i = 2$ an jeder Entwicklungsstelle s_i und der Anzahl der (linear unabhängigen) Eingänge $\tilde{m} = 4$, ergibt sich die Dimension der reduzierten Systeme damit zu $r = \tilde{m} \cdot \hat{i} \cdot \hat{k} = 4 \cdot 5 \cdot 2 = 40$.

Bei der Betrachtung der Approximationsergebnisse der reduzierten Systeme werden die Krylov-Unterraum reduzierten Modelle mit reduzierter Dimension 40 einem modal reduzierten Modell der gleichen Dimension (MODAL_2o) gegenüber gestellt. Die Dimension des modal reduzierten Systems entspricht der Anzahl der Eigenmoden im Intervall $[0 \text{ Hz}, 1122.2 \text{ Hz}]$.

Tabelle A.5: Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 32

Verfahren	Moden	Zeit [s]	Anzahl Iterationen	Fehler $\epsilon_{max}(\mathcal{I})$	Fehler ϵ_{max}
MODAL_2o	32	–	–	73.22	1.68
Verfahren mit Anwendung auf das ungedämpfte System					
MIRA_ND	–	150.66	2	$34.32 \cdot 10^{-3}$	$4.60 \cdot 10^{-3}$
IRGA_ND	–	231.01	2	0.35	0.16
AIRGA_ND	–	431.95	2	376.45	50.48
VAIRGA_ND	–	863.92	2	$6.57 \cdot 10^{-3}$	$0.57 \cdot 10^{-3}$
Verfahren mit Anwendung auf das proportional gedämpfte System					
MIRA_PD	–	212.47	2	$34.32 \cdot 10^{-3}$	$4.59 \cdot 10^{-3}$
IRGA_PD	–	300.51	2	0.35	0.16
AIRGA_PD	–	562.43	2	496.00	79.48
VAIRGA_PD	–	1023.30	2	$84.83 \cdot 10^{-3}$	$79.52 \cdot 10^{-3}$

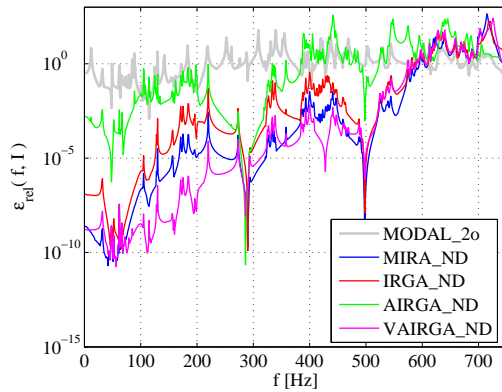
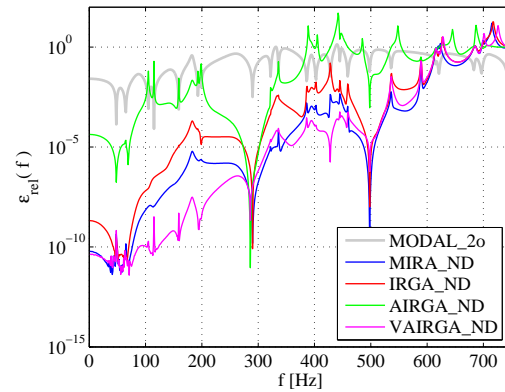

(a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$

(b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung A.9: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND der Dimension 32, angewendet auf das ungedämpfte zweite Beispielmodell

Die Ergebnisse der verschiedenen Reduktionsverfahren sind in der Tabelle A.6 zusammengefasst. Die Abbildungen A.11 und A.12 geben den Verlauf der relativen Fehler (10.3) und (10.5) der verschiedenen reduzierten Modelle wieder.

Approximationsfehler reduzierter Systeme der Dimension 48

Die Algorithmen der Reduktionsverfahren (Algorithmus 19, Algorithmus 21 und Algorithmus 22) werden mit den Anfangsentwicklungsstellen $i \cdot 2\pi$, $100i \cdot 2\pi$, $200i \cdot 2\pi$, $300i \cdot 2\pi$, $400i \cdot 2\pi$ und

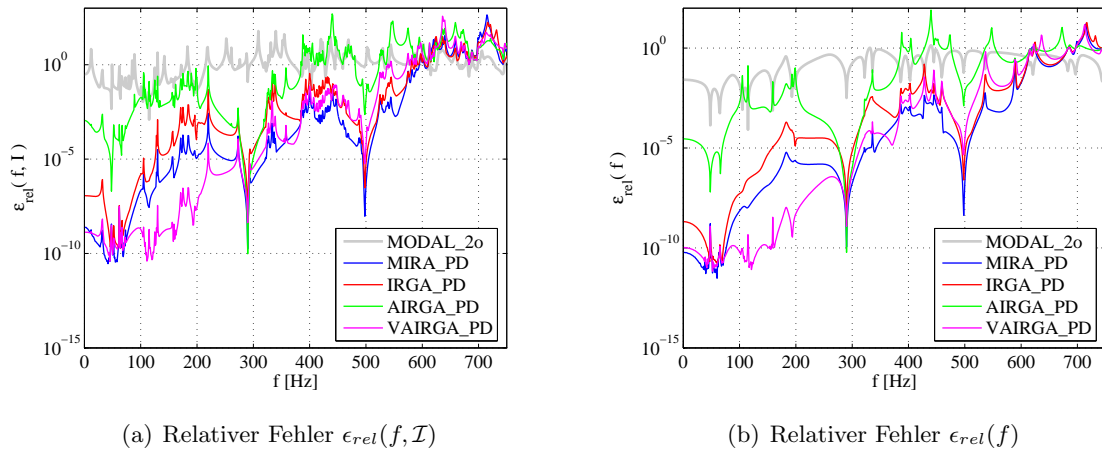


Abbildung A.10: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD der Dimension 32, angewendet auf das proportional gedämpfte zweite Beispielmodell

Tabelle A.6: Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 40

Verfahren	Moden	Zeit [s]	Anzahl Iterationen	Fehler $\epsilon_{max}(\mathcal{I})$	Fehler ϵ_{max}
MODAL_2o	40	—	—	40.50	0.88
Verfahren mit Anwendung auf das ungedämpfte System					
MIRA_ND	—	194.40	2	$1.42 \cdot 10^{-3}$	$0.21 \cdot 10^{-3}$
IRGA_ND	—	283.55	2	$2.08 \cdot 10^{-3}$	$0.25 \cdot 10^{-3}$
AIRGA_ND	—	582.42	2	0.24	$80.38 \cdot 10^{-3}$
VAIRGA_ND	—	1093.70	2	$0.84 \cdot 10^{-3}$	$0.13 \cdot 10^{-3}$
Verfahren mit Anwendung auf das proportional gedämpfte System					
MIRA_PD	—	272.60	2	$1.75 \cdot 10^{-3}$	$0.15 \cdot 10^{-3}$
IRGA_PD	—	379.74	2	$1.35 \cdot 10^{-3}$	$0.21 \cdot 10^{-3}$
AIRGA_PD	—	791.59	2	0.26	$56.54 \cdot 10^{-3}$
VAIRGA_PD	—	1284.10	2	$1.02 \cdot 10^{-3}$	$79.09 \cdot 10^{-6}$

$500i \cdot 2\pi$ (AIRGA_2o) bzw. $\Delta = 0$ (VAIRGA_2o) verwendet. Durch die Anzahl der Anfangsentwicklungsstellen $\hat{i} = 6$, der Stufe des Krylov-Unterraumes $k_i = 2$ an jeder Entwicklungsstelle s_i und der Anzahl der (linear unabhängigen) Eingänge $\tilde{m} = 4$, ergibt sich die Dimension der reduzierten Systeme damit zu $r = \tilde{m} \cdot \hat{i} \cdot \hat{k} = 4 \cdot 6 \cdot 2 = 48$.

Bei der Betrachtung der Approximationsergebnisse der reduzierten Systeme werden die Krylov-Unterraum reduzierten Modelle mit reduzierter Dimension 48 einem modal reduzierten Modell der gleichen Dimension (MODAL_2o) gegenüber gestellt. Die Dimension des modal reduzierten Systems entspricht der Anzahl der Eigenmoden im Intervall $[0 \text{ Hz}, 1264.5 \text{ Hz}]$.

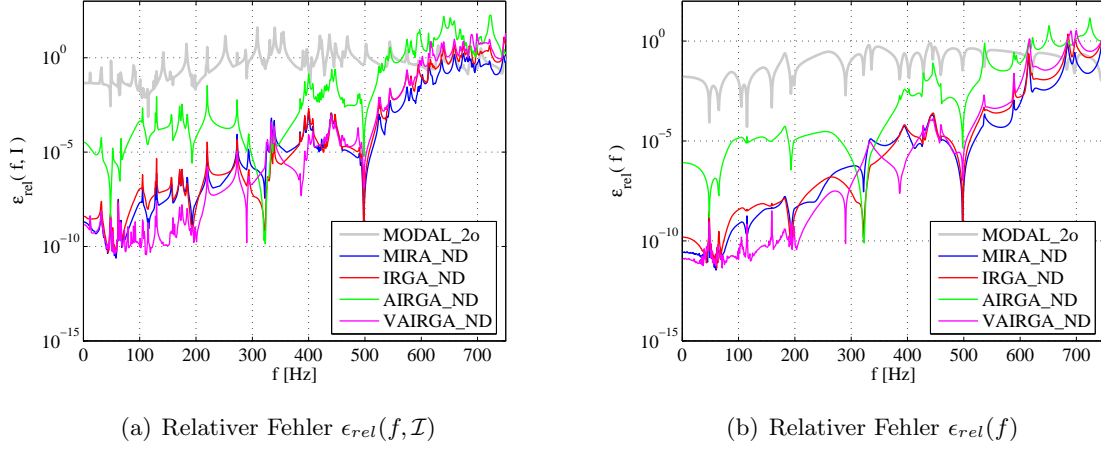


Abbildung A.11: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND der Dimension 40, angewendet auf das ungedämpfte zweite Beispielmodell

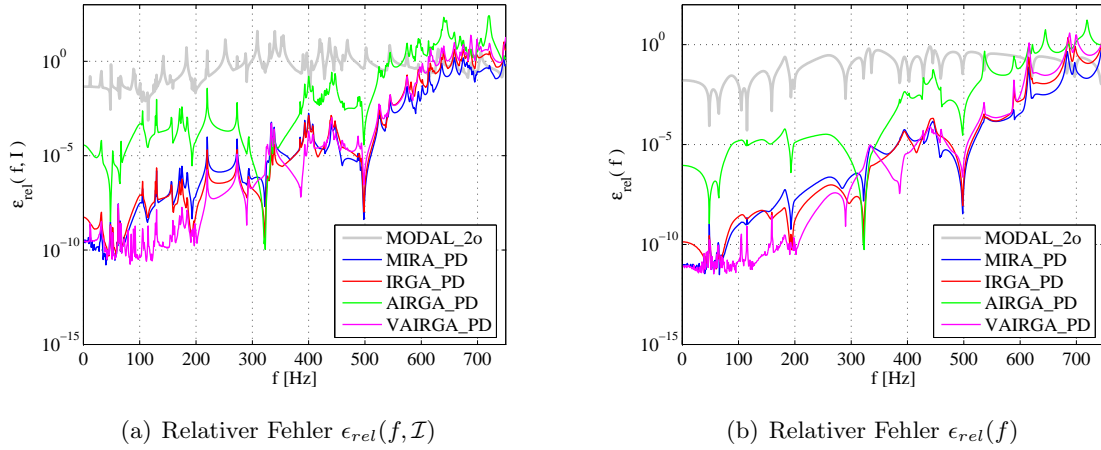


Abbildung A.12: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD der Dimension 40, angewendet auf das proportional gedämpfte zweite Beispielmodell

Die Ergebnisse der verschiedenen Reduktionsverfahren sind in der Tabelle A.7 zusammengefasst. Die Abbildungen A.13 und A.14 geben den Verlauf der relativen Fehler (10.3) und (10.5) der verschiedenen reduzierten Modelle wieder.

Approximationsfehler reduzierter Systeme der Dimension 56

Die Algorithmen der Reduktionsverfahren (Algorithmus 19, Algorithmus 21 und Algorithmus 22) werden mit den Anfangsentwicklungsstellen $i \cdot 2\pi$, $84i \cdot 2\pi$, $167i \cdot 2\pi$, $250.5i \cdot 2\pi$, $332i \cdot 2\pi$, $417i \cdot 2\pi$ und $500i \cdot 2\pi$ sowie $\Delta = 500 \cdot 2\pi$ (AIRGA_2o) bzw. $\Delta = 0$ (VAIRGA_2o) verwendet. Durch die Anzahl der Anfangsentwicklungsstellen $\hat{i} = 7$, der Stufe des Krylov-Unterraumes $k_i = 2$ an jeder

Tabelle A.7: Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 48

Verfahren	Moden	Zeit [s]	Anzahl Iterationen	Fehler $\epsilon_{max}(\mathcal{I})$	Fehler ϵ_{max}
MODAL_2o	48	–	–	39.74	0.82
Verfahren mit Anwendung auf das ungedämpfte System					
MIRA_ND	–	251.85	2	$19.30 \cdot 10^{-6}$	$1.14 \cdot 10^{-6}$
IRGA_ND	–	343.67	2	$1.18 \cdot 10^{-6}$	$61.69 \cdot 10^{-9}$
AIRGA_ND	–	786.98	2	$13.36 \cdot 10^{-3}$	$28.77 \cdot 10^{-6}$
VAIRGA_ND	–	1349.10	2	$14.22 \cdot 10^{-6}$	$0.52 \cdot 10^{-6}$
Verfahren mit Anwendung auf das proportional gedämpfte System					
MIRA_PD	–	337.05	2	$22.10 \cdot 10^{-6}$	$1.20 \cdot 10^{-6}$
IRGA_PD	–	455.61	2	$1.05 \cdot 10^{-6}$	$57.98 \cdot 10^{-9}$
AIRGA_PD	–	1065.90	2	$1.44 \cdot 10^{-3}$	$2.31 \cdot 10^{-6}$
VAIRGA_PD	–	1629.90	2	$8.30 \cdot 10^{-6}$	$0.41 \cdot 10^{-6}$

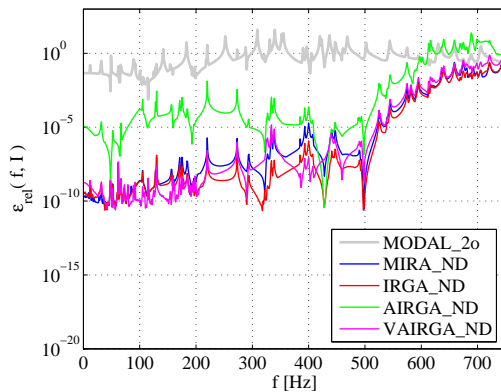
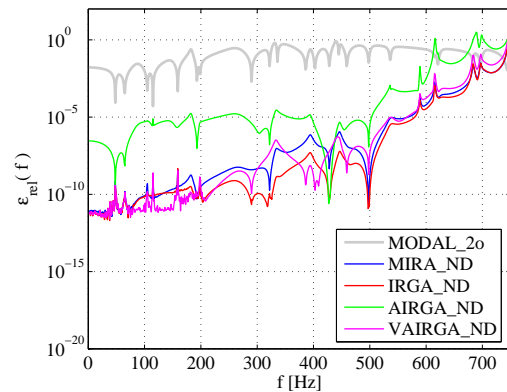
(a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$ (b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung A.13: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND der Dimension 48, angewendet auf das ungedämpfte zweite Beispielmodell

Entwicklungsstelle s_i und der Anzahl der (linear unabhängigen) Eingänge $\tilde{m} = 4$, ergibt sich die Dimension der reduzierten Systeme damit zu $r = \tilde{m} \cdot \hat{i} \cdot \hat{k} = 4 \cdot 7 \cdot 2 = 56$.

Bei der Betrachtung der Approximationsergebnisse der reduzierten Systeme werden die Krylov-Unterraum reduzierten Modelle mit reduzierter Dimension 56 einem modal reduzierten Modell der gleichen Dimension (MODAL_2o) gegenüber gestellt. Die Dimension des modal reduzierten Systems entspricht der Anzahl der Eigenmoden im Intervall $[0 \text{ Hz}, 1469.7 \text{ Hz}]$.

Die Ergebnisse der verschiedenen Reduktionsverfahren sind in der Tabelle A.8 zusammengefasst. Die Abbildungen A.15 und A.16 geben den Verlauf der relativen Fehler (10.3) und (10.5) der

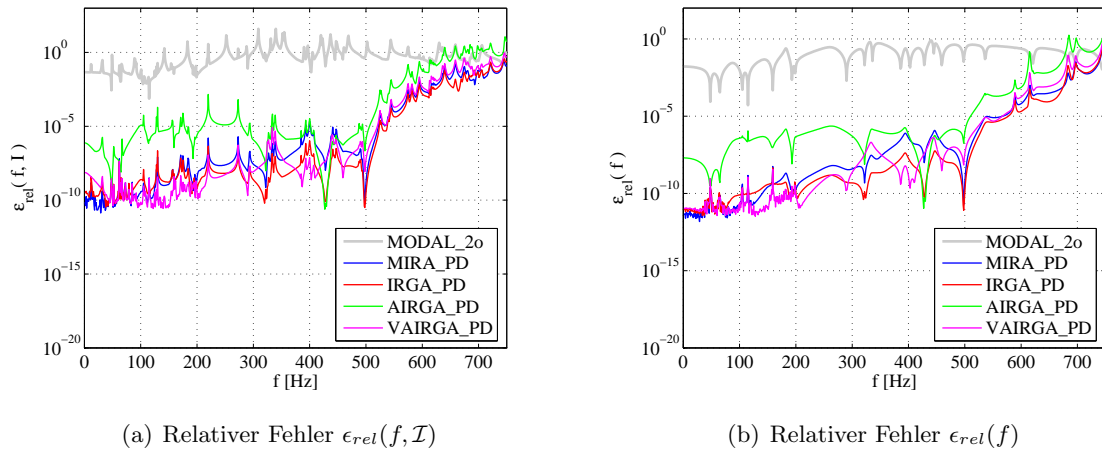


Abbildung A.14: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD der Dimension 48, angewendet auf das proportional gedämpfte zweite Beispielmodell

verschiedenen reduzierten Modelle wieder.

Tabelle A.8: Ergebnisse der untersuchten Reduktionsverfahren - Reduktion auf Dimension 56

Verfahren	Moden	Zeit [s]	Anzahl Iterationen	Fehler $\epsilon_{max}(\mathcal{I})$	Fehler ϵ_{max}
MODAL_2o	56	—	—	39.67	0.81
Verfahren mit Anwendung auf das ungedämpfte System					
MIRA_ND	—	300.27	2	$65.46 \cdot 10^{-6}$	$8.34 \cdot 10^{-6}$
IRGA_ND	—	400.47	2	$2.71 \cdot 10^{-6}$	$0.49 \cdot 10^{-6}$
AIRGA_ND	—	1038.20	2	$3.09 \cdot 10^{-6}$	$65.69 \cdot 10^{-6}$
VAIRGA_ND	—	1663.20	2	$27.71 \cdot 10^{-9}$	$4.76 \cdot 10^{-9}$
Verfahren mit Anwendung auf das proportional gedämpfte System					
MIRA_PD	—	413.15	2	$31.99 \cdot 10^{-6}$	$3.73 \cdot 10^{-6}$
IRGA_PD	—	537.01	2	$5.89 \cdot 10^{-6}$	$0.36 \cdot 10^{-6}$
AIRGA_PD	—	1404.40	2	$6.31 \cdot 10^{-3}$	$66.19 \cdot 10^{-6}$
VAIRGA_PD	—	2051.40	2	$32.21 \cdot 10^{-9}$	$4.15 \cdot 10^{-9}$

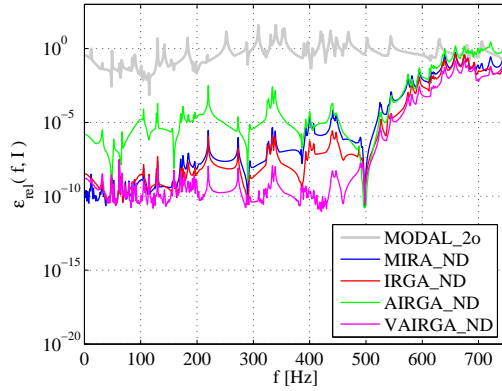
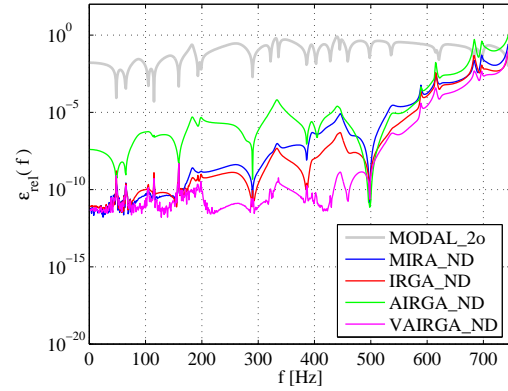
(a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$ (b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung A.15: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_ND, IRGA_ND, AIRGA_ND und VAIRGA_ND der Dimension 56, angewendet auf das ungedämpfte zweite Beispielmodell

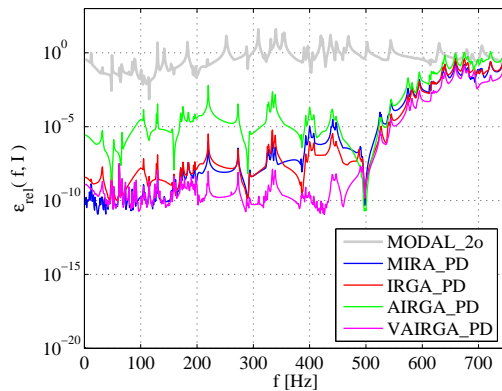
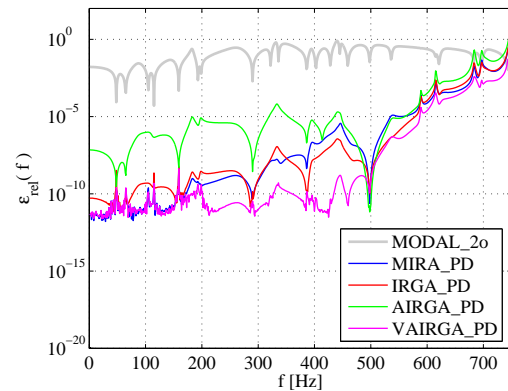
(a) Relativer Fehler $\epsilon_{rel}(f, \mathcal{I})$ (b) Relativer Fehler $\epsilon_{rel}(f)$

Abbildung A.16: Relative Fehler der reduzierten Systeme MODAL_2o, MIRA_PD, IRGA_PD, AIRGA_PD und VAIRGA_PD der Dimension 56, angewendet auf das proportional gedämpfte zweite Beispielmodell

Literaturverzeichnis

- [1] H. Groß, J. Hamann, and G. Wiegärtner, *Elektrische Vorschubantriebe in der Automatisierungstechnik*, Publicis MCD, Erlangen, 2000.
- [2] E. Abele, M. Dervisopoulos, and M. Munirathnam, *Profil eines jungen und dynamischen Marktes - Indien im Fokus der Werkzeugmaschine*, ZWF 101 **11** (2006), 638–642.
- [3] M. Gérardin and D. Rixen, *Mechanical Vibrations*, 2 ed., John Wiley & Sons, Chichester, 1998.
- [4] M. Alam, A. Nieuwoudt, and Y. Massoud, *Provably passive second order model order reduction for package parasitics using spectral zeros*, IEEE Northeast Workshop on Circuits and Systems, NEWCAS, 2007, pp. 791–794.
- [5] M. Alam, A. Nieuwoudt, and Y. Massoud, *Frequency Selective Model Order Reduction via Spectral Zero Projection*, Proceedings of the 2007 Asia and South Pacific Design Automation Conference, 2007, pp. 379–383.
- [6] J.I. Aliaga, D.L. Boley, R.W. Freund, and V. Hern, *A Lanczos-type method for multiple starting vectors*, Mathematics of Computation **69** (2000), 1577–1601.
- [7] H.W. Alt, *Lineare Funktionalanalysis: Eine Anwendungsorientierte Einführung*, vol. 2, Springer, Berlin, 1995.
- [8] Y. Altintas, C. Brecher, M. Weck, and S. Witt, *Virtual Machine Tool*, Anals of the CIRP **54** (2005), 651–673.
- [9] A.C. Antoulas, *A new result on passivity preserving model reduction*, Systems and Control Letters **54** (2005), 361–374.
- [10] ———, *Approximation of Large-Scale Dynamical Systems*, SIAM, Philadelphia, USA, 2005.
- [11] W.E. Arnoldi, *The principle minimized iteration in the solution of the matrix eigenproblem*, Quaterly of Applied Mathematics **9** (1951), 17–29.
- [12] Z. Bai, P. Feldmann, and R. W. Freund, *Stable and Passive Reduced-Order Models Based on Partial Padé Approximation via the Lanczos Process*, Numerical Analysis Manuscript Nr. 97/3-10, Bell Laboratories, Murray Hill, New Jersey (1997).

-
- [13] Z. Bai, R.D. Slone, W.T. Smith, and Q. Ye, *Error Bound for Reduced System Model by Padé Approximation via the Lanczos Process*, IEEE Transactions on Circuits and Systems **18** (1999), 133–141.
- [14] Z. Bai and Y. Su, *Dimension Reduction of Large-Scale Second-Order Dynamical Systems via a Second-Order Arnoldi Method*, SIAM Journal on Scientific Computing **26** (2005), 1692–1709.
- [15] ———, *SOAR: A Second-order Arnoldi Method for the Solution of the Quadratic Eigenvalue Problem*, SIAM Journal on Matrix Analysis and Applications **26** (2005), no. 3, 640–659.
- [16] Z. Bai and Q. Ye, *Error estimation of the Padé approximation of transfer functions via the Lanczos process*, IEEE Transactions on Circuits and Systems **7** (1998), 1–17.
- [17] L. Baratchart, M. Cardelli, and M. Olivi, *Identification and rational \mathcal{L}_2 approximation: A gradient algorithm*, Automatica **27** (1991), no. 2, 413–418.
- [18] K.-J. Bathe, *Finite Element Procedures*, Prentice-Hall, Upper Saddle River, 1996.
- [19] U. Baur and P. Benner, *Factorized Solutions of Lyapunov Equations Based on Hierarchical Matrix Arithmetic*, Computing **78** (2006), no. 3, 211–234.
- [20] C. Beattie and S. Gugercin, *Krylov-based model reduction of second-order systems with proportional damping*, Proceedings of the 44th IEEE Conference on Decision and Control, 2005, pp. 2278–2283.
- [21] T. Bechtold, *Model Order Reduction of Electro-Thermal MEMS*, Dissertation, Universität Freiburg, Freiburg, 2005.
- [22] T. Bechtold, E.B. Rudnyi, and J.G. Korvink, *Error Estimation for Arnoldi-based Model Order Reduction of MEMS*, 2004.
- [23] P. Benner, *Solving large-scale control problems*, IEEE Control Systems Magazine **14** (2004), no. 1, 44–59.
- [24] P. Benner, V. Mehrmann, and D.C. Sorensen, *Dimension Reduction of Large-Scale Systems, volume 45 of: Lectures in Computational Science and Engineering*, Springer, Berlin/Heidelberg, 2005.
- [25] P. Benner and J. Saak, *Efficient Balancing based MOR for Second Order Systems Arising in Control of Machine Tools*, Proceedings of the 2009 Conference of Mathematical Modelling, 2009, pp. 1232–1243.
- [26] J. Berkemer, *Gekoppelte Simulation von Maschinendynamik und Antriebsregelung unter Verwendung linearer Finite Elemente Modelle*, Dissertation, Universität Stuttgart, Stuttgart, 2003.

-
- [27] G. Bianchi, F. Paolucci, P. Van den Braembussche, and H. Van Brussel, *Towards Virtual Engineering in Machine Tool Design*, Annals of the CIRP **45** (1996), no. 1, 381–384.
- [28] T. Bonin, A. Soppa, J. Saak, M. Zaeh, H. Faßbender, and P. Benner, *Moderne Modellordnungsreduktionsverfahren für Finite-Elemente-Modelle zur Simulation von Werkzeugmaschinen*, Erscheint in: Proceedings of the VDI-Mechatroniktagung 2011, 31.03-01.04.2011, Dresden (2011), 6 Seiten.
- [29] G. Brandenburg, *Trends in der elektrischen Antriebstechnik*, A&D Kompendium (2000), 36–37.
- [30] C. Brecher, *Vergleichende Analyse von Vorschubantrieben für Werkzeugmaschinen (Berichte aus der Produktionstechnik 17/2002)*, Shaker, Aachen, 2002.
- [31] S. Brückl, *Hochgenaue Regelung von Linearmotoren durch optimierte Strommessung*, Antriebstechnik **38** (1999), no. 9, 90–93.
- [32] A. Bunse-Gerstner, D. Kubalińska, G. Vossen, and D. Wilczek, *H_2 -norm optimal model reduction for large-scale discrete dynamical MIMO systems*, Journal of Computational and Applied Mathematics **233** (2009), no. 5, 1202–1216.
- [33] R. Bürgel, *Prozessanalyse an spanenden Werkzeugmaschinen mit digital geregelten Antrieben*, iwb Forschungsberichte 151, Herbert Utz, München, 2001.
- [34] F.M. Callier and C.A. Desoer, *Linear System Theory*, Springer, London, UK, 1991.
- [35] T.K. Caughey, *Classical Normal Modes in Damped Linear Dynamic Systems*, Journal of Applied Mechanics, June (1960) (1960), 269–271.
- [36] V.-S. Chellaboina and W.M. Haddad, *Is the Frobenius norm induced?*, IEEE Transactions on Automatic Control **40** (1995), 2137–2139.
- [37] V.-S. Chellaboina, W.M. Haddad, D.S. Bernstein, and D.A. Wilson, *Induced convolution norms of linear dynamical systems*, Mathematics of Control Signals and Systems **13** (2000), 216–239.
- [38] E. Chiprout and M.S. Nakhla, *Asymptotic Waveform Evaluation*, Kluwer Academic, Dordrecht, 1994.
- [39] C.-C. Chu, M.H. Lai, and W.S. Feng, *MIMO Interconnects Order Reductions by Using the Multiple Point Adaptive-Order Rational Global Arnoldi Algorithm*, IEICE Transactions on Electronics **E89-C(6)** (2006), 792–802.
- [40] ———, *The multiple point global Lanczos method for multiple-inputs multiple-outputs interconnect order reduction*, IEICE Transactions on the Fundamentals of Electronics, Communication and Computer Science **E89-A(10)** (2006), 2706–2716.
- [41] ———, *Lyapunov-based error estimations of MIMO interconnect reductions by using the global Arnoldi algorithm*, IEICE Transactions on Electronics **E90-A(2)** (2007), 415–418.

-
- [42] ———, *Model-order reductions for MIMO systems using global Krylov subspace methods*, Mathematics and Computers in Simulation **79** (2008), 1153–1164.
- [43] C.-C. Chu, H.-J. Lee, and W.S. Feng, *Error Estimation of Arnoldi-Based Interconnect Model-Order Reductions*, IEICE Transactions on the Fundamentals of Electronics, Communication and Computer Science **E88-A** (2005), no. 2, 533–537.
- [44] C.K. Chui and G. Chen, *Linear Systems and Optimal Control*, Springer, Berlin, 1989.
- [45] R.L. Clark, *Accounting for Out-of-Bandwidth Modes in the Assumed Modes Approach: Implications on Colocated Output Feedback Control*, Transactions of the ASME **119** (1997), 390–395.
- [46] R.R. Craig, *Structural Dynamics*, John Wiley & Sons, New York, 1981.
- [47] R.R. Craig and Z. Ni, *Component Mode Synthesis for Modal Order Reduction of Nonclassically Damped Systems*, AIAA Journal **12** (1989), no. 4, 577–584.
- [48] J. Cullum, A. Ruehli, and T. Zhang, *A method for reduced-order modeling and simulation of large interconnect circuits and its application to PEEC models including retardation*, IEEE Trans. on Circuits and Systems-II: Analog and Digital signal processing **47** (2000), no. 4, 261–273.
- [49] C. de Villemagne and R.E. Skelton, *Model Reduction Using a Projection Formulation*, International Journal of Control **46** (1987), 2141–2169.
- [50] S. Dietz, *Vibration and Fatigue Analysis of Vehicle Systems Using Component Modes*, VDI Fortschritt-Berichte, Reihe 12, Nr. 401, VDI-Verlag, Düsseldorf, 1999.
- [51] P. Van Dooren, K.A. Gallivan, and P.-A. Absil, *\mathcal{H}_2 -optimal model reduction of MIMO systems*, Applied Mathematics Letters **21** (2008), no. 12, 1267–1273.
- [52] G.H.A. Dötsch, *Einführung in Theorie und Anwendung der Laplace-Transformation: Ein Lehrbuch für Studierende der Mathematik, Physik und Ingenieurwissenschaft*, Birkhäuser, Englewood Cliffs, USA, 1958.
- [53] P.L. Duren, *Theory of \mathcal{H}^p spaces*, Academic Press, New York, 1970.
- [54] R. Eid, B. Salimbahrami, B. Lohmann, E.B. Rudnyi, and J.G. Korvink, *Parametric Order Reduction of Proportionally Damped Second Order Systems*, Technical Report on Automatic Control TRAC-1, Technische Universität München, Lehrstuhl für Regelungstechnik (2006).
- [55] I.M. Elfadel and D.D. Ling, *A block rational Arnoldi algorithm for multipoint passive model-order reduction of multiport RLC networks*, Proceedings of the 1997 IEEE/ACM International Conference on Computer-Aided Design, 1997, pp. 66–71.
- [56] P. Eubert, *Digitale Zustandsregelung elektrischer Vorschubantriebe*, Springer, Berlin, 1992.

-
- [57] H. Faßbender and A. Soppa, *Machine Tool Simulation based on Reduced Order FE Models*, Proceedings of the 2009 Conference of Mathematical Modelling, 2009, pp. 1266–1277.
- [58] ———, *Machine Tool Simulation based on Reduced Order FE Models*, Accepted in: Mathematics and Computers in Simulation (2010).
- [59] P. Feldmann, *Model order reduction techniques for linear systems with large numbers of terminals*, Proceedings of the Conference on Design, Automation and Test in Europe, 2004, pp. 44–47.
- [60] P. Feldmann and R. W. Freund, *Efficient linear circuit analysis by Padé approximation via the Lanczos process*, IEEE Transactions in Computer-Aided Design **14** (1995), 639–649.
- [61] P. Feldmann and R.W. Freund, *Reduced-Order Modeling of Large Linear Subcircuits via a Block Lanczos Algorithm*, Proceedings of the 32nd annual ACM/IEEE Design Automation Conference, 1995, pp. 474–479.
- [62] P. Feldmann and F. Liu, *Sparse and efficient reduced order modeling of linear subcircuits with large number of terminals*, Proceedings of the International Conference on Computer-Aided Design, 2004, pp. 88–92.
- [63] O. Föllinger, *Regelungstechnik - Einführung in die Methoden und ihre Anwendung*, 8 ed., Hüthig, Heidelberg, 1994.
- [64] M. Frangos and I. M. Jaimoukha, *Adaptive rational interpolation: Arnoldi and Lanczos-like equations*, European Journal of Control (2008), 342–354.
- [65] F.D. Freitas, J. Rommes, and N. Martins, *Gramian-Based Reduction Method Applied to Large Sparse Power System Descriptor Models*, IEEE Transactions on Power Systems **23** (2008), no. 3, 1258–1270.
- [66] R. W. Freund, *Passive reduced-order models for interconnect simulation and their computation via Krylov-subspace algorithms*, Proceedings of the 1999 IEEE Design Automation Conference, 1999, pp. 195–200.
- [67] ———, *Model Reduction Methods Based on Krylov Subspaces*, Acta Numerica **12** (2003), 267–319.
- [68] R.W. Freund, *SPRIM: Structure-Preserving Reduced-Order Interconnect Macromodeling*, Proceedings of the 2004 IEEE/ACM International conference on Computer-aided design, 2004, pp. 80–87.
- [69] P. Fulcheri and M. Olivi, *Matrix Rational \mathcal{H}_2 Approximation: A Gradient Algorithm based on Schur Analysis*, SIAM Journal on Control and Optimization **36** (1998), no. 6, 2103–2127.
- [70] K. Gallivan, E. Grimme, D. Sorensen, and P. Van Dooren, *On some modifications of the Lanczos algorithm and the relation with Padé approximations*, 1995.

-
- [71] K. Gallivan, E.J. Grimme, and P. van Dooren, *A Rational Lanczos aAlgorithm for Model Reduction*, Numerical Algorithms **12** (1996), 33–63.
- [72] F.R. Gantmacher, *Matrizentheorie, Band 86 der Reihe Hochschulbücher für Mathematik*, vol. 2, VEB Deutscher Verlag der Wissenschaften, 1986.
- [73] R. Gasch and K. Knothe, *Strukturdynamik, Band 2: Kontinua und ihre Diskretisierung*, Springer, Berlin, 1989.
- [74] W.K. Gawronski, *Dynamics and Control of Structures*, Springer, Berlin, Germany, 1998.
- [75] ———, *Advanced Structural Dynamics and Active Control of Structures*, Springer, Berlin, Germany, 2004.
- [76] K. Glover, *All optimal Hankel-norm approximation of linear multivariable systems and their L^∞ -error bounds*, International Journal of Control **39** (1984), 1115–1193.
- [77] G.H. Golub and C.F. van Loan, *Matrix Computations*, John Hopkins University Press, 1996.
- [78] E. Griepentrog and R. März, *Differential-algebraic equations and their numerical treatment, Band 88 der Reihe Teubner-Texte zur Mathematik*, Teubner, Leipzig, 1986.
- [79] ———, *Differential-algebraic equations and their numerical treatment, Band 88 der Reihe: Teubner-Texte zur Mathematik*, Teubner Verlagsgesellschaft, Leipzig, 1986.
- [80] E. Grimme, K. Gallivan, and P. van Dooren, *A Rational Lanczos Algorithm for Model Reduction II: Interpolation Point Selection*, Technical report, University of Illinois at Urbana Champaign (1998).
- [81] E.J. Grimme, *Krylov projection methods for model reduction*, PhD Thesis, University of Illinois at Urbana-Champaign, Urbana-Champaign, 1997.
- [82] E.J. Grimme, D. Sorensen, and P. Van Dooren, *Model Reduction of State Space Systems via an Implicitly Restarted Lanczos Method*, Numerical Algorithms **12** (1995), 1–31.
- [83] H. Gronbach, *Werkzeugmaschinenentwicklung benötigt Simulation - Auf dem Weg zur virtuellen Maschine*, In: G. Reinhard: Mechatronische Produktionssysteme - Genauigkeit gezielt entwickeln, Herbert Utz, München, 2002.
- [84] S. Gugercin, *Projection methods for model reduction of large-scale dynamical systems*, PhD Thesis, ECE Department, Rice University, Houston, December, 2005.
- [85] S. Gugercin and A.C. Antoulas, *An \mathcal{H}_2 error expression for the Lanczos procedure*, Proceedings of the 2003 IEEE Conference on Decision and Control, 2003, pp. 1869–1872.
- [86] ———, *A Survey of Model Reduction by Balanced Truncation and Some New Results*, International Journal of Control **77** (2004), 748–766.

-
- [87] S. Gugercin, A.C. Antoulas, and C. Beattie, *\mathcal{H}_2 -Model Reduction for Large-Scale Linear Systems*, SIAM Journal on Matrix Analysis and Applications **30** (2008), no. 2, 609–638.
- [88] R.J. Guyan, *Reduction of Stiffness and Mass Matrices*, AIAA Journal **3** (1965), no. 2, 380.
- [89] Y. Halevi, *Frequency weighted model reduction via optimal projection*, Proceedings of the 1999 IEEE Conference on Decision and Control, 1999, pp. 2906–2911.
- [90] M. Heyouni, *The global Hessenberg and CMRH methods for linear systems with multiple right-hand sides*, Numerical Algorithms **26** (2001), no. 4, 317–332.
- [91] N.J. Higham, D.S. Mackey, N. Mackey, and F. Tisseur, *Symmetric linearizations for matrix polynomials*, SIAM Journal on Matrix Analysis and Applications **29** (2006), no. 1, 143–159.
- [92] D. Hinrichsen and A.J. Pritchard, *Mathematical Systems Theory I. Modelling, State Space Analysis, Stability and Robustness, Volume 48 of Texts in Applied Mathematics*, Springer, Berlin, Heidelberg, 2005.
- [93] R.M. Hintz, *Analytical Methods in Component Modal Synthesis*, AIAA Journal **13** (1975), 1007–1016.
- [94] K. Hoffman, *Banach spaces of analytic functions*, Prentice Hall, London, 1962.
- [95] R.A. Horn and C.R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, 1991.
- [96] R. Ionutiu, J. Rommes, and A.C. Antoulas, *Passivity-Preserving Model Reduction Using Dominant Spectral-Zero Interpolation*, IEEE Transactions on Automatic Control **27** (2008), 2250–2263.
- [97] R. Isermann, *Digitale Regelsysteme, Band II*, 2 ed., Springer, Berlin, 1987.
- [98] ———, *Digitale Regelsysteme, Band I*, 2 ed., Springer, Berlin, 1988.
- [99] J. Jablonowski, *The 2007 Worl Machine-Tool Output & Consumption Survey. Metalworking Insiders' Report, January 31, 2007*, Gardner Publications, Inc. (2007).
- [100] I.M. Jaimoukha and E.M. Kasenally, *Oblique Projection Methods for Large Scale Model Reduction*, SIAM Journal on Matrix Analysis and Applications **16** (1995), 602–627.
- [101] ———, *Implicitly Restarted Krylov Subspace Methods For Stable Partial Realizations*, SIAM Journal on Matrix Analysis and Applications **18** (1997), 633–652.
- [102] K. Jbilou, A. Messaoudi, and H. Sadok, *Global FOM and GMRES algorithms for matrix equations*, Applied Numerical Mathematics **31** (1999), 49–63.
- [103] K. Jbilou and A.J. Riquet, *Projection methods for large Lyapunov matrix equations*, Linear Algebra and its Applications **415** (2006), no. 2-3, 344–358.
- [104] T. Kailath, *Linear Systems*, Prentice-Hall, Englewood Cliffs, USA, 1980.

-
-
- [105] M. Kamon, F. Wang, and J. White, *Generating nearly optimally compact models from Krylov-subspace based reduced-order models*, IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing **47** (2000), 239–248.
- [106] G. Kerschen, J.-P. Golinval, A.F. Vakakis, and L.A. Bergman, *The Method of Proper Orthogonal Decomposition for Dynamical Characterization and Order Reduction of Mechanical Systems: An Overview*, Nonlinear Dynamics **41** (2005), 147–169.
- [107] P. Kirchknopf, *Ermittlung modaler Parameter aus Übertragungsfrequenzgängen*, Springer, Berlin, 1989.
- [108] H.W. Knobloch and H. Kwakernaak, *Lineare Kontrolltheorie*, Springer, Berlin, Heidelberg, 1985.
- [109] A. Köhler, *Modellreduktion von linearen Deskriptorsystemen erster und zweiter Ordnung mit Hilfe von Block-Krylov-Unterraumverfahren*, Diplomarbeit, Technische Universität Bergakademie Freiberg, Freiberg (2006).
- [110] P. Koosis, *Introduction to \mathcal{H}^p spaces*, Cambridge University Press, Cambridge, 1980.
- [111] D. Kubalińska, *Optimal interpolation-based model reduction*, Dissertation, Universität Bremen, Bremen, 2008.
- [112] F. Küçükay, *Dynamik der Zahnradgetriebe. Modelle, Verfahren, Verhalten*, Springer, Berlin, 1987.
- [113] M.-H. Lai, C.-C. Chu, and W.-S. Feng, *Applications of AOGL Model-Order Reduction Techniques in Interconnect Analysis*, Proceedings of the 2007 IEEE International Symposium on Circuits and Systems, 2007, pp. 1133–1136.
- [114] C. Lanczos, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, Journal of research of the National Bureau of Standards **45** (1950), 225–280.
- [115] A. Laschet, *Simulation von Antriebssystemen*, Springer, Berlin, 1988.
- [116] H.-J. Lee, C.-C. Chu, and W.-S. Feng, *An adaptive-order rational Arnoldi method for model-order reductions of linear time-invariant systems*, Linear Algebra and its Applications **415** (2004), 235–261.
- [117] M. Lehner, *Modellreduktion in elastischen Mehrkörpersystemen*, Dissertation, Universität Stuttgart, Stuttgart, 2007.
- [118] M. Lehner and P. Eberhard, *On the Use of Moment-Matching to Build Reduced Order Models in Flexible Multibody Dynamics*, Multibody System Dynamics **16** (2006), 191–211.
- [119] ———, *A Two-Step Approach for Model Reduction in Flexible Multibody Dynamics*, Multibody System Dynamics **17** (2007), 157–176.

-
- [120] W. Leonhard, *Control on Electrical Drives*, 2 ed., Springer, Berlin, 1996.
- [121] J.-R. Li, *Model Reduction of Large Linear Systems via Low Rank System Gramians*, PhD Thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2000.
- [122] R.-D. Li and Z. Bai, *Structure-Preserving Model Reduction Using a Krylov Subspace Projection Formulation*, Communications in Mathematical Sciences **3** (2005), 179–199.
- [123] B. Liu, X. Zeng, Y. Su, J. Tao, Z. Bai, C. Chiang, and D. Zhou, *Block SAPOR: Block Second-order Arnoldi method for Passive Order Reduction of multi-input multi-output RCS interconnect circuits*, Proceedings of the 2005 Asia and South Pacific Design Automation Conference, 2005, pp. 244–249.
- [124] L. Meier and D.G. Luenberger, *Approximation of linear constant systems*, IEEE Transactions on Automatic Control **12** (1967), 585–588.
- [125] R. Lozano, B. Brogliato, O. Egeland, and B. Maschke, *Dissipative Systems Analysis and Control. Communications and Control Engineering*, Springer, London, 2000.
- [126] D.G. Luenberger, *Optimization by Vector Space Methods*, John Wiley & Sons, New York, 1969.
- [127] D.S. Mackey, N. Mackey, C. Mehl, and V. Mehrmann, *Vector spaces of linearizations for matrix polynomials*, SIAM Journal on Matrix Analysis and Applications **28** (2006), no. 4, 971–1004.
- [128] K. Magnus and K. Popp, *Schwingungen*, 5 ed., Teubner, Stuttgart, 1997.
- [129] J.H. Mathews and R.E. Howell, *Complex Analysis for Mathematics and Engineering*, Jones and Barlett Publishers, USA, 2006.
- [130] L. Meirovitch, *Dynamics and Control of Structures*, Wiley, New York, 1990.
- [131] L. Meirovitch and M.K. Kwak, *Convergence of the Classical Rayleigh-Ritz Method and the Finite Element Method*, AIAA Journal **28** (1990), 1509–1516.
- [132] D.G. Meyer and S. Srinivasan, *Balancing and Model Reduction for Second-Order Form Linear Systems*, IEEE Transactions on Automatic Control **41** (1996), 1632–1644.
- [133] B.C. Moore, *Principal component analysis in linear systems: controllability, observability and model reduction*, IEEE Transactions on Automatic Control **26** (1981), no. 1, 17–32.
- [134] P.C. Müller, *Stabilität und Matrizen*, Springer, Berlin, 1977.
- [135] P.C. Müller and W.O. Schiehlen, *Lineare Schwingungen*, Akademische Verlagsgesellschaft, Wiesbaden, 1976.
- [136] T.V. Nguyen and J. Li, *Multipoint Padé Approximation Using a Rational Block Lanczos Algorithm*, Proceedings of the 1997 IEEE/ACM International Conference on Computer-Aided Design, 1997, pp. 72–75.

-
-
- [137] M. Odabasioglu, M. Celik, and T. Pileggi, *Prima: Passive reduced-order interconnect macromodeling algorithm*, IEEE Transactions on CAD of Integrated Circuits and Systems **17** (1998), no. 8, 645–654.
 - [138] T. Oertli, *Strukturmechanische Berechnung und Regelungssimulation von Werkzeugmaschinen mit elektromechanischen Vorschubantrieben*, Dissertation, Technische Universität München, München, 2008.
 - [139] B. Oestereich, *Analyse und Design mit UML 2: Objektorientierte Softwareentwicklung*, 7 ed., Oldenbourg, München, 2005.
 - [140] K.H.A. Olsson, *Model Order Reduction in FEMLAB by Dual Rational Arnoldi*, Thesis for the degree of licentiate of engineering, Chalmers University of Technology and Göteborg University, Göteborg, 2002.
 - [141] K.H.A. Olsson and A. Ruhe, *Rational Krylov for Eigenvalue Computation and Model Order Reduction*, BIT Numerical Mathematics **46** (2006), 99–111.
 - [142] H. Padé, *Sur la représentation approchée d’une fonction par des fractions rationnelles*, Ann. Ec. Norm. Sup. **9** (1892), 1–93.
 - [143] T. Penzl, *A cyclic low rank Smith method for large sparse Lyapunov equations*, SIAM Journal on Scientific Computing **21** (2000), no. 4, 1401–1418.
 - [144] ———, *Algorithms for Model Reduction of Large Dynamical Systems*, Linear Algebra and its Applications **415** (2006), no. 2-3, 322–343.
 - [145] F. Pfeiffer, *Einführung in die dynamik*, Teubner, Stuttgart, 1992.
 - [146] T. Reis and T. Stykel, *Balanced truncation model reduction of second-order systems*, Mathematical and Computer Modelling of Dynamical Systems **14** (2008), no. 5, 391–406.
 - [147] U. Riefenstahl, *Elektrische Antriebstechnik*, Teubner, Stuttgart, Leipzig, 2000.
 - [148] M. Rosenblum and J. Rovnyak, *Hardy classes and Operator theory*, Oxford University Press, Oxford, 1985.
 - [149] ———, *Topics in hardy classes and univalent functions*, Oxford University Press, Oxford, 1994.
 - [150] W.J. Rugh, *Linear System Theory*, vol. 2, Prentice Hall, Upper Saddle River, USA, 1996.
 - [151] A. Ruhe, *The rational Krylov algorithm for nonsymmetric eigenvalue problems. III: Complex shifts for real matrices*, BIT Numerical Mathematics **34** (1994), 165–176.
 - [152] Y. Saad, *Iterative Methods for Sparse Linear Systems*, vol. 2, SIAM, Philadelphia, 2003.
 - [153] M. Sadkane, *Block Arnoldi and Davidson methods for unsymmetric large eigenvalue problems*, Numerical Mathematics **64** (1993), 195–211.

-
-
- [154] B. Salimbahrami and B. Lohmann, *Stopping Criterion in Order Reduction of Large Scale Systems Using Krylov Subspace Methods*, Proceedings in Applied Mathematics and Mechanics (PAMM) **4** (2004), no. 1, 682–683.
 - [155] S.B. Salimbahrami, *Structure Preserving Order Reduction of Large Scale Second Order Models*, Dissertation, Technische Universität München, München, 2005.
 - [156] S.B. Salimbahrami and B. Lohmann, *Order reduction of large scale second-order systems using Krylov subspace methods*, Linear Algebra and its Applications **415** (2006), no. 2-3, 385–405.
 - [157] G. Schelfhout, *Model reduction for control design*, PhD Thesis, Univerisite Catholique De Louvain, Louvain, 1996.
 - [158] C. Schneider, *Strukturmechanische Berechnung in der Werkzeugmaschinenkonstruktion*, *iwb Forschungsberichte*, Herbert Utz, München, 2000.
 - [159] R. Schönfeld, *Bewegungssteuerungen - Digitale Signalverarbeitung, Drehmomentsteuerung, Bewegungsablaufsteuerung, Simulation*, Springer, Berlin, 1998.
 - [160] D. Schröder, *Elektrische Antriebe 2 - Regelung von Antrieben*, Springer, Berlin, 1995.
 - [161] R. Schwertassek and O. Wallrapp, *Dynamik flexibler Mehrkörpersysteme*, Vieweg, Braunschweig, 1999.
 - [162] L.F. Shampine, M.W. Reichelt, and J.A.Kierzenka, *Solving Index-1 DAEs in MATLAB and Simulink*, SIAM Review **41** (1999), 538–552.
 - [163] W. Simon, *Elektrische Vorschubantriebe an NC-Systemen*, *iwb Forschungsberichte* 5, Springer, Berlin, 1986.
 - [164] J.-J. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall, Englewood Cliffs, 1991.
 - [165] E.D. Sontag, *Mathematical control theory: deterministic systems, volume 6 of Texts in Applied Mathematics*, Springer, New York, USA, 1990.
 - [166] ———, *Mathematical Control Theory*, vol. 2, Springer, Berlin, 1998.
 - [167] D.C. Sorensen, *Passivity preserving model reduction via interpolation of spectral zeros*, Systems and Control Letters **54** (2005), 347–360.
 - [168] J.T. Spanos, M.H. Milman, and D.L. Mingori, *A new algorithm for \mathcal{L}_2 optimal model reduction*, Automatica (1992), 897–909.
 - [169] G.W. Stewart, *Matrix Algorithms, Volume II: Eigensystems*, SIAM, Philadelphia, USA, 2001.
 - [170] T. Stykel, *Analysis and Numerical Solution of Generalized Lyapunov Equations*, Dissertation, Technische Universität Berlin, Berlin, 2002.

-
-
- [171] ———, *Gramian based model reduction for descriptor systems*, Mathematics of Control, Signals, and Systems **16** (2004), 297–319.
 - [172] ———, *Balanced Truncation Model Reduction of Second-Order Systems*, Proceedings of the 5th Conference of Mathematical Modelling Vienna, February 8-10, 2006.
 - [173] T.-J. Su and R.R. Craig, *Model Reduction and Control of Flexible Structures Using Krylov Vectors*, Journal of Guidance, Control, and Dynamics **14** (1991), 260–267.
 - [174] Y. Su, J. Wang, X. Zeng, Z. Bai, C. Chiang, and D. Zhou, *SAPOR: Second-order Arnoldi method for passive order reduction of RCS circuits*, Proceedings of the 2004 IEEE/ACM International conference on Computer-aided design, 2004, pp. 74–79.
 - [175] H. Summer, *Modell zur Berechnung verzweigter Antriebsstrukturen*, Springer, Berlin, 1986.
 - [176] S.X.-D. Tan and L. He, *Advanced Model Order Reduction Techniques in VLSI Design*, Cambridge University Press, Cambridge, United Kingdom, 2007.
 - [177] F. Tisseur and K. Merbergen, *A survey of the quadratic eigenvalue problem*, SIAM Review **43** (2001), 234–286.
 - [178] A. Vandendorpe, *Model reduction of linear systems, an interpolation point of view*, PhD Thesis, Universite Catholique De Louvain, Louvain, 2004.
 - [179] S. Volkwein, *Model Reduction Using Proper Orthogonal Decomposition*, Vorlesungsmanuskript. Karl-Franzens-Universität Graz. Institut für Mathematik und Wissenschaftliches Rechnen, 2006.
 - [180] G. Vossen, A. Bunse-Gerstner, D. Kubalinska, and D. Wilczek, *Necessary optimality conditions for \mathcal{H}_2 -norm optimal model reduction. Technical report, University of Bremen*, 2007.
 - [181] W. Walter, *Einführung in die Theorie der Distributionen*, BI Wissenschaftsverlag, Zürich, 1974.
 - [182] M. Weck and C. Brecher, *Werkzeugmaschinen 3 - Mechatronische Systeme, Vorschubantriebe, Prozessdiagnose*, Springer, Berlin, 2006.
 - [183] A. Wick, *Synchroner Drehstrom-Servoantrieb mit Transistor-Pulsumrichter*, Dissertation, Universität Erlangen-Nürnberg, Nürnberg, 1982.
 - [184] J. Wittenburg, *Dynamik of Systems of Rigid Bodies*, Teubner, Stuttgart, 1977.
 - [185] W.S. Yoo and E.J. Haug, *Dynamics of Articulated Structures. Part I: Theory*, Journal of Structural Mechanics **14** (1986), 105–126.
 - [186] M.F. Zäh, *Megatrends in der Produktion - Herausforderungen für Gesellschaft, Wissenschaft und Ausbildung*, In: Zäh, M.F. u. a. (Hrsg.): Grenzen überwinden - Wachstum der neuen Art. Tagungsband zum Münchner Kolloquium, München, 27.02.2003, Herbert Utz, München, 2003.

-
-
- [187] M.F. Zäh and T. Oertli, *FEM Analysis and Control System Simulation of Ball Screw Feed Drive Systems in Machine Tools*, In: Proceedings of the NAFEMS Seminar “Mechatronics in Structural Analysis”, Wiesbaden, Germany, Paper No. 9, 2004.
- [188] K. Zhou, J.C. Doyle, and K. Glover, *Robust and Optimal Control*, Prentice Hall, Upper Saddle River, 1996.
- [189] O.C. Zienkiewicz, *The finite element method in engineering science*, Hanser, München, Wien, 1975.
- [190] O.C. Zienkiewicz and R.L. Taylor, *The Finite Element Method*, 4 ed., vol. 2, McGraw-Hill, London, 1991.

